



TUGAS AKHIR - KS 141501

**RANCANG BANGUN PROTOTIPE FITUR VOTING
SENSITIF WAKTU PADA HASIL PENCARIAN
PERANGKAT LUNAK UNTUK MENINGKATKAN
RELEVANSI**

***DEVELOPMENT OF TIME-SENSITIVE VOTING
FEATURE PROTOTYPE ON SOFTWARE PRODUCT
SEARCH RESULT TO IMPROVE SEARCH RELEVANCE***

AHMAD YUSUF FAUZAN ADIMAH
NRP 5213 100 133

Dosen Pembimbing:
Rully Agus Hendrawan, S.Kom, M.Eng
Andre Parvian Aristio, S.Kom, M.Sc

DEPARTEMEN SISTEM INFORMASI
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember
Surabaya 2017



ITS
Institut
Teknologi
Sepuluh Nopember

TUGAS AKHIR - KS 141501

**RANCANG BANGUN PROTOTYPE FITUR VOTING
SENSITIF WAKTU PADA HASIL PENCARIAN
PERANGKAT LUNAK UNTUK MENINGKATKAN
RELEVANSI**

**DEVELOPMENT OF TIME-SENSITIVE VOTING FEATURE
PROTOTYPE ON SOFTWARE PRODUCT SEARCH
RESULT TO IMPROVE SEARCH RELEVANCE**

**AHMAD YUSUF FAUZAN ADIMAH
NRP 5213 100 133**

**Dosen Pembimbing:
Rully Hendrawan, S.Kom, M.Eng
Andre Parvian Aristio, S.Kom, M.Sc**

**DEPARTMENT OF INFORMATION SYSTEM
FACULTY OF INFORMATION TECHNOLOGY
Sepuluh Nopember Institute of Technology
Surabaya 2017**

LEMBAR PENGESAHAN

RANCANG BANGUN PROTOTIPE FITUR VOTING SENSITIF WAKTU PADA HASIL PENCARIAN PERANGKAT LUNAK UNTUK MENINGKATKAN RELEVANSI

TUGAS AKHIR

Disusun untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada
Departemen Sistem Informasi
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember

Oleh:



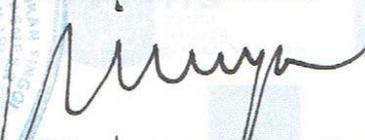
AHMAD YUSUF FAUZAN ADIMAH

NRP 5213 100 133

Surabaya, Juni 2017



**KEPALA
DEPARTEMEN SISTEM INFORMASI**


Dr. Ir. Aris Tjahyanto, M.Kom.
NIP 19650310 199102 1 001

LEMBAR PERSETUJUAN

RANCANG BANGUN PROTOTIPE FITUR VOTING SENSITIF WAKTU PADA HASIL PENCARIAN PERANGKAT LUNAK UNTUK MENINGKATKAN RELEVANSI

TUGAS AKHIR

Disusun untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada
Departemen Sistem Informasi
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember

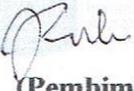
Oleh:

AHMAD YUSUF FAUZAN ADIMAH

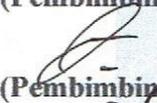
NRP 5213 100 133

Disetujui Tim Penguji: Tanggal Ujian : 07 Juli 2017
Periode Wisuda : September 2017

Rully Hendrawan S.Kom, M.Eng


(Pembimbing 1)

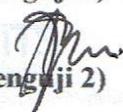
Andre Parvian Aristio S.Kom, M.Sc


(Pembimbing 2)

Mahendrawati ER, ST, M.Sc, Ph.D


(Penguji 1)

Erma Suryani, ST, MT, Ph.D


(Penguji 2)

**RANCANG BANGUN PROTOTIPE FITUR VOTING
SENSITIF WAKTU PADA HASIL Pencarian
PERANGKAT LUNAK UNTUK MENINGKATKAN
RELEVANSI**

Nama Mahasiswa : Ahmad Yusuf Fauzan Adimah
NRP : 5213100133
Jurusan : Sistem Informasi FTif – ITS
Dosen Pembimbing 1 : Rully Hendrawan S.Kom, M.Eng
Dosen Pembimbing 2 : Andre Parvian Aristio S.Kom,
M.Sc

ABSTRAK

Fitur search merupakan sumber terbesar untuk memudahkan akses informasi di internet. Untuk mendapatkan pelanggan baru maka sebuah website harus menyajikan konten yang tepat dalam waktu yang cepat sehingga mampu meningkatkan jumlah pelanggan dan bersaing dengan website lainnya.

Namun sering kali pelanggan tidak menemukan aplikasi yang benar-benar ia butuhkan saat melakukan pencarian. Dalam layanan konten digital seperti google play aplikasi yang memiliki rating paling tinggi berada pada posisi paling atas, dan aplikasi yang memiliki voting lebih sedikit akan mendapatkan rating yang lebih tinggi dibandingkan dengan aplikasi yang memiliki lebih banyak voting. Sehingga pelanggan lebih sering mendownload aplikasi yang berada pada urutan paling atas padahal belum tentu aplikasi tersebut merupakan aplikasi yang paling relevan dengan kebutuhannya.

Pada penelitian ini, akan dibangun sebuah prototipe fitur voting dengan menggunakan perhitungan Wilson Score Interval. Metode ini mempunyai sifat yang baik untuk perhitungan jumlah kecil dan jumlah besar atau ekstrim sehingga sesuai untuk diimplementasikan dalam studi kasus

ini. Dari penelitian yang dilakukan didapatkan hasil bahwa prototipe yang dikembangkan dengan menggunakan metode Wilson Score Interval dapat meningkatkan relevansi hasil pencarian yang sesuai dengan perspektif pengguna. Hal ini dikarenakan jumlah voting pengguna pada suatu aplikasi diperhitungkan sehingga urutan yang ditampilkan cocok dengan hasil voting yang sudah dilakukan dan urutan aplikasi sesuai dengan apa yang pengguna nilai.

Kata Kunci: Prototipe, Website, Voting, Average Rating, Wilson Score Interval

**DEVELOPMENT OF TIME-SENSITIVE VOTING
FEATURE PROTOTYPE ON SOFTWARE PRODUCT
SEARCH RESULT TO IMPROVE SEARCH
RELEVANCE**

Name : Ahmad Yusuf Fauzan Adimah
NRP : 5213100133
Department : Sistem Informasi FTIf – ITS
Supervisor 1 : Rully Hendrawan S.Kom, M.Eng
Supervisor 2 : Andre Parvian Aristio S.Kom, M.Sc

ABSTRACT

The search feature is the largest source for easy access to information on the internet. To get new customers then a website should present the right content in the fast that is able to increase the number of customers and compete with other websites

But often the customers do not find applications that really he needed when searching. In digital content services like google play application that has the highest rating being at the top position, and applications have fewer voting would get a higher rating than applications that have more voting. So that customers more often download applications that are at the top of the order but not necessarily such an application is the application that is most relevant to their needs.

In this research, will be built a prototype voting feature by using the calculation of the Wilson Score Interval. This method has good properties for a calculation of the number of small and large quantities or extreme so according to be implemented in this case study. From research conducted obtained results that are developed using the prototype method of the Wilson Score Interval can improve the relevance of search results that match the user's perspective.

This is due to the number of users voting on an application is taken into account so that the displayed order matches the results of the voting that has already been done and the order of the applications in accordance with what the user value.

Keywords: Prototype, Website, Voting, Average Rating, Wilson Score Interval

KATA PENGANTAR

Puji dan syukur kehadiran Allah SWT yang telah memberi kekuatan dan hikmat sehingga penulis dapat menyelesaikan penulisan skripsi yang berjudul

“Rancang Bangun Prototipe Fitur Voting Sensitif Waktu Pada Hasil Pencarian Perangkat Lunak Untuk Meningkatkan Relevansi”

Skripsi ini merupakan tugas akhir akademik sebagai salah satu syarat untuk menyelesaikan pendidikan pada Departemen Sistem Informasi, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember. Penulis menyadari bahwa dalam menyelesaikan skripsi ini tidak lepas dari bantuan berbagai pihak, sehingga skripsi ini dapat selesai dengan baik dan tepat waktu. Oleh karena itu pada kesempatan ini penulis ingin mengucapkan terima kasih kepada:

1. Allah SWT yang senantiasa melimpahkan berkah dan rahmat-Nya selama penulis mengerjakan Tugas Akhir.
2. Nabi Muhammad SAW yang telah membimbing penulis menuju jalan yang dirahmati Allah SWT.
3. Bapak Ir. Aris Tjahyanto, M.Kom. selaku Ketua Jurusan Sistem Informasi ITS Surabaya.
4. Bapak Rully Agus Hendrawan S.Kom, M.Eng. dan Bapak Andre Parvian Aristio S.Kom, M.Sc. selaku dosen pembimbing Tugas Akhir penulis yang telah memberikan banyak pengetahuan dan pemahaman baru bagi penulis.
5. Bapak Rully Agus Hendrawan S.Kom, M.Eng. selaku dosen wali penulis yang memberikan motivasi sehingga penulis terus mengusahakan yang terbaik selama mengerjakan Tugas Akhir.
6. Bapak, Ibu dan Kakak tercinta yang telah mengajarkan hal berharga dan menjadi inspirasi serta motivasi penulis untuk selalu memberikan yang terbaik dalam Tugas Akhir ini.

7. Saudara dan kerabat tercinta yang senantiasa mendoakan kelancaran serta kesuksesan dalam pengerjaan Tugas Akhir.
8. Saudari Fitri Larasati Rifqatusa'adah Lubis terkasih yang telah membantu dan memotivasi dalam pengerjaan Tugas Akhir ini.
9. Teman – teman “MAFIA” yang telah memberikan banyak pengalaman, cerita, dan kenangan selama penulis melakukan penelitian dan di waktu-waktu lainnya.
10. Pri Rezki Destrianto, Umar Al Aqsho, Marina Safitri, Harun Rizal dan Stezar Priansyah yang telah membantu penulis dalam menyelesaikan Tugas Akhir ini.
11. Kawan – Kawan Lab Sistem Enterprise (SE) yang menjadi rekan senasib dan seperjuangan.
12. Penghuni Lab ADDI yang telah mempersilakan penulis bernaung dan mencari inspirasi dalam mengerjakan Tugas Akhir ini.
13. Seluruh teman – teman, Keluarga Beltranis yang selalu memberikan dukungan kepada penulis dan memberikan banyak cerita selama penulis melakukan studi.

Semoga tulisan ini bermanfaat bagi orang yang membaca, bagi penelitian dan pengembangan aplikasi di masa depan. Penulis menyadari bahwa laporan ini masih jauh dari kesempurnaan karena kesempurnaan sejatinya hanya milik Allah SWT, maka saran dan kritik yang konstruktif dari semua pihak sangat diharapkan demi perbaikan selanjutnya.

Surabaya, 04 Juli 2017

Penulis

DAFTAR ISI

ABSTRAK.....	iii
ABSTRACT.....	v
KATA PENGANTAR	vii
DAFTAR ISI.....	ix
DAFTAR GAMBAR	xiii
DAFTAR TABEL.....	xv
1 BAB I PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Perumusan Masalah.....	3
1.3 Batasan Masalah.....	3
1.4 Tujuan Penelitian.....	3
1.5 Manfaat Penelitian.....	4
1.6 Relevansi Tugas Akhir	4
2 BAB II TINJAUAN PUSTAKA	5
2.1 Penelitian Sebelumnya	5
2.2 Dasar Teori	9
2.2.1 Google Play Store.....	10
2.2.2 Web Scrapping	10
2.2.3 Customer Relationship Management (CRM).....	12
2.2.4 Teknologi Web.....	13
2.2.5 MySQL.....	17
2.2.6 Wilson Score Interval	17
2.2.7 Average Rating.....	17

2.2.8 Perbandingan Average Rating Dengan Wilson Score Interval	18
3 BAB III METODOLOGI	21
3.2 Uraian Metodologi Penelitian	22
3.2.1 Studi Literatur	22
3.2.2 Inisiasi Kebutuhan	22
3.2.2 Pengembangan Prototipe	22
3.2.3 Penyusunan Buku Tugas Akhir	23
4 BAB IV PERANCANGAN PENELITIAN	25
4.1 Perancangan Pembuatan Protipe	25
4.1.1 Perancangan Pengambilan data	25
4.1.1.1 Identifikasi Karakteristik Situs	25
4.1.2 Desain Basis Data	30
4.1.2.1 Perancangan Basis Data Konseptual	30
4.1.2.1.1 Identifikasi Tipe Entity	30
4.1.2.1.2 Identifikasi Tipe Relasi	30
4.1.2.1.3 Identifikasi Dan Menghubungkan Atribut Dengan Entiti Atau pun Dengan Tipe Relasi	31
4.1.2.1.4 Menentukan Domain Atribut	32
4.1.2.1.5 Menentukan Atribut Candidate Key Dan Primary	32
4.1.2.1.6 Mempertimbangkan Penggunaan Konsep Pemodelan Yang Lebih Baik	33
4.1.2.1.7 Memeriksa Dengan Model Redundansi	33
4.1.2.1.8 Data Dictionary	34
4.1.2.2 Perancangan Basis Data Logikal	40

4.1.2.2.1 Menentukan Relasi Untuk Model Data Logikal	40
4.1.2.2.1.1 Mapping of Strong Entity Types.....	40
4.1.2.2.1.2 Mapping of Weak Entity Types	42
4.1.2.2.1.3 Mapping of Binary (1:1) Relationship Types	43
4.1.2.2.1.4 Mapping of Binary (1:N) Relationship Types	43
4.1.2.2.1.5 Mapping of Binary (M:N) Relationship Types	43
4.1.2.2.1.6 Mapping of Multivalued Attributes	44
4.1.2.2.1.7 Mapping of N-ary Relationship Types.	44
4.1.2.2.1.8 Mapping of Specialization or Generalization	44
4.1.2.3 Perancangan Basis Data Fisikal	44
4.1.2.3.2 Merancang Relasi Dasar	44
4.1.2.3.1 Menerjemahkan Model Data Logikal Ke Target DBMS	45
4.2 Rancang Bangun Prototipe	46
4.2.1 User Interface	47
4.2.2 Use Case Diagram	47
4.2.2.1 Daftar Use Case	48
4.2.3 Descriptive Case.....	48
4.3 Perancangan Validasi Prototipe	Error! Bookmark not defined.
5 BAB V IMPLEMENTASI	51
5.1 Lingkungan Implementasi	51

5.2 Struktur Direktori	52
5.3 Pengambilan Data.....	53
5.4 Rancang Bangun Prototipe	60
5.4.1 Implementasi Halaman Prototipe Fitur Voting	60
6 BAB VI HASIL DAN PEMBAHASAN	69
6.1 Hasil Prototipe Aplikasi	69
6.1.1 Halaman Fitur Voting.....	69
6.2 Hasil Analisis Perbandingan.....	70
6.2.1 Average Rating Dengan Wilscon Score Interval (Real Time).....	70
6.2.2 Eksperimen Desain Berdasarkan User Review dan User Experience.....	78
6.2.3 Google Play Store Dengan Prototipe Fitur Voting	87
7 BAB VII KESIMPULAN DAN SARAN.....	95
7.1 Kesimpulan	95
7.2 Saran	97
DAFTAR PUSTAKA.....	99
BIODATA PENULIS.....	103
LAMPIRAN A.....	105
LAMPIRAN B	115
LAMPIRAN C	119

DAFTAR GAMBAR

Gambar 2.1. Visualisasi Effect dari Submission Time [3].....	6
Gambar 2.2. Visualisasi Logarithm Scale [3]	6
Gambar 2.3. Visualisasi Tanpa Logarithm Scale [3]	7
Gambar 2.4. Visualisasi Efek Down Vote [3]	7
Gambar 2.5. Visualisasi Web Scrapping [10].....	12
Gambar 2.6. Evaluasi Web [16]	14
Gambar 2.7. Layer Semantic Web.....	16
Gambar 2.8. Penerapan Average Rating [21]	18
Gambar 2.9. Visualisasi Wilson Score Interval [3]	19
Gambar 3.1. Metodologi penelitian	21
Gambar 4. 1. Alur Pengambilan Data Dari API JSON	26
Gambar 4. 2. Alur Pengambilan Data Dari Proses API JSON	26
Gambar 4. 3. Entitas Play Store.....	29
Gambar 4. 4. Relasi Entitas	31
Gambar 4. 5. Relasi Antar Entitas	31
Gambar 4. 6. Menentukan Domain Atribut	32
Gambar 4. 7. Proses Generalisasi Pada Playstore	33
Gambar 4. 8 Entitas Googleplay_scrap_list	40
Gambar 4.9 MRD Googleplay_scrap_list	41
Gambar 4.10 Entitas Googleplay_scrap_app.....	41
Gambar 4.11 MRD Googleplay_scrap_app.....	41
Gambar 4.12 Mapping Weak Entity	42
Gambar 4.13 MRD Mapping Weak Entity	43
Gambar 4. 14 Mapping Binary (1:1)	43
Gambar 4. 15. Relasi Applications dan Tes.....	45
Gambar 4.16 User Interface Prototipe	47
Gambar 4. 17. Use Case Diagram	48
Gambar 5. 1. Struktur Direktori	52
Gambar 6.1 Halaman Fitur Voting	69
Gambar 6.2 Tampilan hasil pencarian aplikasi kategori musik pada Googleplay	88

Gambar 6.3 Perbandingan Aplikasi Music Player dan Spotify	89
Gambar 6.4 Tampilan aplikasi musik pada Prototipe Fitur Voting	90
Gambar A1 Route Sistem Pada Aplikasi	105
Gambar A.2 Kode Tabel Pada Aplikasi.....	105
Gambar A.3 Kode Untuk Menampilkan Tabel	106
Gambar A.4 Kode Untuk Tombol Action Up dan Down	106
Gambar A.5 Kode Untuk CSS home.blade.php.....	107
Gambar A.6 Tabel Pada Aplikasi	108
Gambar A.7 Kode Untuk Menampilkan Tabel	108
Gambar A.8 Kode Penulisan Rumus Wilson Score Interval	109
Gambar A.9 Action Untuk Tombol Up	110
Gambar A.10 Action Untuk Tombol Down	111
Gambar A.11 Controller Untuk Menampilkan Halaman Home	112
Gambar A.12 Untuk Mengupdate Database Ketika Tombol Up Ditekan	112
Gambar A.13 Untuk Mengupdate Database Ketika Tombol Down Ditekan	112
Gambar C.1 Hasil Review Perspektif Pengguna.....	119
Gambar C.2 Alasan Pengguna Bila Tidak Setuju	119
Gambar C.3 Hasil Review Pengguna Faktor Knowledge	119
Gambar C.4 Hasil Review Perspektif Pengguna Dari Kedua Metode	120
Gambar C.5 Hasil Review Perspektif Pengguna Ketepatan Hasil Urutan Ranking.....	120
Gambar C.6 Hasil Review Perspektif Pengguna Prototipe dengan GooglePlay	121
Gambar C.7 Hasil Review Perspektif Pengguna Ketepatan Hasil Urutan Ranking.....	121

DAFTAR TABEL

Tabel 1.1. Relevansi Tugas Akhir	4
Tabel 2.1. Penelitian Sebelumnya	8
Tabel 2.2. Perbandingan Average Rating & Wilson Score Interval	19
Tabel 4. 1. Deskripsi Data Atribut Pada Tabel Applications	27
Tabel 4. 2. Data Dictionaru Pada Applications	36
Tabel 4. 3. Daftar Use Case	48
Tabel 4. 4. Use Case Point 1.	48
Tabel 4. 5. Use Case Point 2	49
Tabel 4. 6. Use Case Point 3	49
Tabel 4.7 Use Case Point 4	49
Tabel 5. 1. Spesifikasi Perangkat Lunak	51
Tabel 5.2 Kode Untuk Konfigurasi Koneksi Database Googleplayapi	53
Tabel 5.3 Kode Untuk Lanjutan Proses Pengambilan Data ...	54
Tabel 5.4 Kode Untuk Proses Pengelolaan Data	55
Tabel 5.5 Kode Untuk Melakukan Input Data	56
Tabel 5.6 Kode Untuk Proses Pengambilan Data	57
Tabel 5.7 Kode Untuk Proses Konversi Data	57
Tabel 5.8 Kode Untuk Proses Pemasukan Data Hasil Konversi	58
Tabel 5.9 Kode Untuk Menampilkan Nama Tabel Pada Halaman Fitur Voting	60
Tabel 5.10 Implementasi Wilson Score Interval	62
Tabel 5.11 Kode Untuk Menghitung Jumlah Voting Up	63
Tabel 5.12 Kode Untuk Menghitung Jumlah Voting Down ..	64
Tabel 5.13 Kode Untuk Menampilkan Hasil Voting	65
Tabel 5.14 Kode Untuk Menampilkan Data Dari Basis Data ..	66
Tabel 5.15 Kode Untuk Melakukan Perhitungan Up	67
Tabel 5.16 Kode Untuk Melakukan Perhitungan Down	68

Tabel 6. 1 Hasil Perhitungan Average Rating dan Wilson Score Interval	70
Tabel 6. 2 Rangkaing Aplikasi Berdasarkan Kedua Metode ...	74
Tabel 6. 3 Hasil Voting Kedua Metode Berdasarkan Review Aplikasi (Fase Pertama Kedua Metode)	79
Tabel 6.4 Hasil Perhitungan Average Rating Pada Fase Pertama dan Kedua (User Experience).....	81
Tabel 6.5 Hasil Perhitungan Wilson Score Interval Pada Fase Pertama dan Kedua (User Experience).....	83
Tabel 6. 6 Rangkaing Aplikasi dari kedua metode	84
Tabel 6.7 Rangkaing Aplikasi pada GooglePlay dan Prototipe Fitur Voting.....	91
Tabel B.1 Bukti Voting Pada Fase Pertama dan Kedua.....	115

BAB I

PENDAHULUAN

Bab ini akan menjelaskan proses identifikasi masalah antara lain latar belakang permasalahan, perumusan masalah, batasan masalah, tujuan dari pembuatan tugas akhir, manfaat yang di dapat dari tugas akhir, serta relevansi tugas akhir terhadap permasalahan yang ingin diselesaikan.

1.1 Latar Belakang

Perkembangan teknologi di Indonesia semakin pesat salah satunya adalah penggunaan internet. Hingga akhir tahun 2016 jumlah pengguna internet di Indonesia telah mencapai lebih dari 52 juta pengguna[1]. Hal ini membuktikan bahwa pengguna internet di Indonesia cukup tinggi. Pengguna internet biasanya melakukan aktivitas seperti browsing info terkini, social media, chatting, video streaming, bermain game, dan juga lifestyle. Tidak hanya melalui internet, aktivitas diatas dapat dilakukan dengan menggunakan aplikasi mobile dimana aplikasi yang diinginkan dapat di download pada google play dan apps store. Sampai saat ini kedua media download tersebut merupakan media download terbesar[2]. Untuk meningkatkan loyalitas pelanggan yang disediakan google play maka seharusnya google play memperhatikan kebutuhan dari pelanggan terkait layanan yang disediakan.

Namun saat ini terlihat bahwa layanan yang disediakan belum sepenuhnya sesuai dengan apa yang dibutuhkan pelanggan. Sering kali pelanggan tidak menemukan aplikasi yang benar-benar ia butuhkan saat melakukan pencarian. Dapat dilihat aplikasi yang memiliki rating paling tinggi berada pada posisi paling atas, namun biasanya user tidak memperhatikan rating yang dimiliki aplikasi tersebut apakah valid atau tidak. Biasanya aplikasi yang memiliki voting lebih sedikit akan mendapatkan rating yang lebih tinggi dibandingkan dengan aplikasi yang memiliki lebih banyak voting. Sehingga

pelanggan lebih sering mendownload aplikasi yang berada pada urutan paling atas padahal belum tentu aplikasi tersebut aplikasi tersebut merupakan aplikasi yang paling sesuai dengan kebutuhannya.

Beberapa website menggunakan menggunakan metode perhitungan average rating karena lebih mudah dan praktis dalam implementasinya. Metode ini digunakan untuk menghitung rating dari sebuah item atau produk aplikasi kemudian dilakukan perangkingan agar mempermudah pelanggan dalam menemukan produk atau aplikasi yang sesuai dengan keinginannya. Average rating dapat bekerja dengan baik jika jumlah voting yang didapatkan dalam jumlah banyak, akan tetapi kelamahannya terdapat pada saat jumlah voting yang didapatkan kecil, maka perhitungan yang dihasilkan menjadi berbias. Jika perhitungan yang dihasilkan menjadi berbias maka akan berpengaruh terhadap perangkingan produk dan aplikasi yang ditampilkan ternyata tidak sesuai dengan apa yang diinginkan pelanggan. Hal ini bisa diatasi jika kita menggunakan Wilson score interval karena dengan menggunakan metode ini perhitungan dengan jumlah kecil dan jumlah besar atau ekstrim sekalipun dapat dihitung dengan baik. Dengan menggunakan confidence level sebesar 95% dimana hal ini tidak ada di average rating. Jika perhitungan rating yang dilakukan mampu dihasilkan dengan baik maka perangkingan juga akan dihasilkan dengan baik sehingga produk yang ditampilkan sesuai dengan apa yang diinginkan pelanggan. Nantinya berdasarkan perhitungan metode ini akan didapatkan rating dan ranking yang lebih sesuai dan dapat meningkatkan relevansi hasil pencarian produk.

Untuk menangani permasalahan tersebut maka dalam penelitian ini akan akan dibangun pengembangan prototipe fitur voting pada hasil pencarian produk. Dimana prototipe yang dibangun

berbasis website. Prototipe ini akan menampilkan aplikasi yang berada di google play.

Dengan adanya prototipe ini dapat memudahkan pengguna untuk aplikasi yang paling sesuai dengan kebutuhannya. Urutan yang ditampilkan berdasarkan rating yang telah dilakukan perhitungan menggunakan metode Wilson Score Interval. Sehingga aplikasi yang dibutuhkan dapat meningkatkan relevansi pada hasil pencarian..

1.2 Perumusan Masalah

Pada kondisi saat ini produk tampil dengan urutan acak atau produk yang menurut pelanggan penting posisinya tidak selalu berada diatas dan tidak selalu relevan. Akibatnya pelanggan susah menemukan kembali produk yang menurut dia anggap penting sehingga pengalaman terhadap website e-commerce tersebut berkurang, customer tidak ingin kembali dan beralih ke situs lain.

1.3 Batasan Masalah

Berdasarkan rumusan masalah di atas, batasan masalah dalam tugas akhir ini dapat diatur sebagai berikut:

1. Sumber data yang digunakan didapatkan dari aplikasi yang berada pada google play
2. Voting hanya akan dilakukan pada 100 aplikasi yang berada pada google play.
3. Prototipe yang dirancang berfokus untuk menampilkan hasil perankingan yang paling relevan.

1.4 Tujuan Penelitian

Tujuan daripada tugas akhir ini adalah:

1. Membangun prototipe fitur voting dan ranking pada website pengelolaan koleksi aplikasi mobile menggunakan metode perankingan Wilson Score Interval

2. Meningkatkan relevansi hasil pencarian produk dengan mempertimbangkan masukan (voting) pengguna.

1.5 Manfaat Penelitian

Berikut merupakan manfaat yang diperoleh bagi pengguna website direktori aplikasi:

1. Meningkatkan pengalaman pengunjung web dengan menampilkan aplikasi yang paling relevan dengan kebutuhan user.
2. Memberikan kemudahan kepada pengguna dalam menemukan aplikasi yang relevan.

1.6 Relevansi Tugas Akhir

Relevansi tugas akhir akan membahas mata kuliah yang terkait dengan topik penelitian tugas akhir serta keterkaitan dengan roadmap laboratorium Sistem Enterprise Jurusan Sistem Informasi ITS. Mata kuliah yang berkaitan dengan topik tugas akhir ini telah dipetakan berdasarkan rumusan masalah, seperti yang ditunjukkan pada Tabel 1.

Tabel 1.1. Relevansi Tugas Akhir

No.	Rumusan Masalah	Mata Kuliah
1.	Bagaimana proses pencarian produk dilakukan?	Desain Basis Data, Manajemen Basis Data, Integrasi Aplikasi Korporasi
2.	Bagaimana prototipe yang dihasilkan dengan menggunakan metode Wilson Score Interval?	Pemrograman Berbasis Web, Konstruksi Pengembangan Perangkat Lunak

BAB II

TINJAUAN PUSTAKA

Bab ini akan menjelaskan mengenai penelitian sebelumnya dan dasar teori yang dijadikan acuan atau landasan dalam pengerjaan tugas akhir ini. Landasan teori akan memberikan gambaran secara umum dari landasan penjabaran tugas akhir ini.

2.1 Penelitian Sebelumnya

Dalam penelitian tugas akhir ini, ada beberapa penelitian terdahulu yang dapat dijadikan sebagai bahan kajian studi literatur. Adapun penelitian sebelumnya mengenai gambaran umum penelitian, tujuan penelitian, hasil penelitian, dan keterkaitan penelitian akan diulas pada bab ini. Berikut adalah penelitian terdahulu terkait dengan tugas akhir ini.

Reddit merupakan aplikasi web tentang berita social amerika, web content rating, dan situs diskusi. Pada bagian ini akan difokuskan bagaimana Reddit meranking story dan comment [3]. Berikut merupakan bagaimana cara Reddit melakukan perangkaian story.

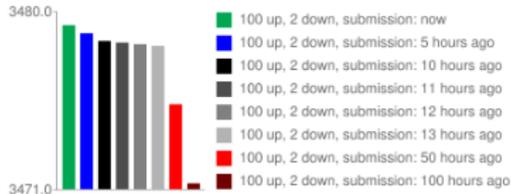
Effect dari Submission Time

Terdapat dua poin yang perlu diperhatikan dalam perangkaian Reddit story ranking, yaitu:

- Submission Time mempunyai dampak besar dalam perangkaian dan algoritma akan meranking cerita baru lebih tinggi daripada cerita lama

- Skor tidak akan menurun seiring berjalannya waktu, tapi cerita baru akan mendapatkan skor lebih tinggi daripada cerita lama

Berikut merupakan contoh visualiasasi skor dari cerita yang mempunyai upvotes dan downvotes yang sama, namun memiliki waktu submission yang berbeda:



Gambar 2.1. Visualisasi Effect dari Submission Time [3]

The logarithm scale

Reddit hot ranking menggunakan fungsi logarithm scale dimana nilai pemilih pertama lebih tinggi daripada sisanya.

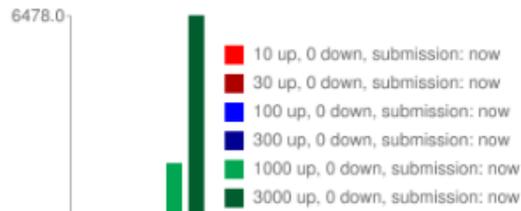
- 10 upvotes pertama memiliki nilai yang sama dengan 100 upvotes berikutnya yang dimana mempunyai nilai yang sama dengan 1000 upvotes berikutnya dan seterusnya

Berikut merupakan contoh visualisasi dari Logarithm Scale:



Gambar 2.2. Visualisasi Logarithm Scale [3]

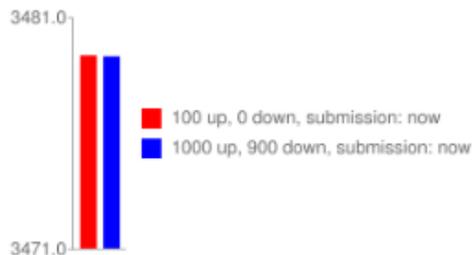
Berikut merupakan contoh apabila tidak menggunakan fungsi logarithm scale, maka skor akan tampak seperti berikut:



Gambar 2.3. Visualisasi Tanpa Logarithm Scale [3]

Efek Downvotes

Reddit merupakan satu dari beberapa situs yang mempunyai fungsi downvotes. Berikut visualisasi dari Efek Downvotes:



Gambar 2.4. Visualisasi Efek Down Vote [3]

Dari penjelasan yang telah dipaparkan diatas maka kesimpulan dari Reddit Story Ranking adalah:

- Submission time merupakan parameter yang sangat penting, sehingga nilai cerita baru akan lebih tinggi daripada cerita lama

- 10 upvotes pertama memiliki nilai yang sama dengan 100 upvotes berikutnya, contoh sebuah cerita memiliki 10 upvotes dan sebuah cerita memiliki 50 upvotes maka kedua cerita akan memiliki peringkat yang sama
- Cerita kontroversial yang mendapatkan upvotes dan downvotes dengan nilai yang sama akan mendapat peringkat lebih rendah daripada cerita yang hanya mendapatkan upvotes.

Tabel 2.1. Penelitian Sebelumnya

Dipublikasikan pada, Tahun	<i>Internasional Journal On Web Service Computing, 2012</i>
Judul	<i>AN INTEGRATED RANKING ALGORITHM FOR EFFICIENT INFORMATION COMPUTING IN SOCIAL NETWORKS[4]</i>
Nama	R. Suri; Pushpa
Tujuan	Tujuan dari penelitian ini adalah melakukan perangkingan pada social network dimana factor popularitas dikombinasikan dengan rangking parsial untuk memperkirakan peringkat global untuk objek web yang dicari
Hasil	Hasil pada penelitian ini secara keseluruhan menunjukkan bahwa kerangka Fusion Based Ranking dapat menghasilkan pencarian yang lebih baik daripada traditional rangking dalam hal relevansi dan metode FBR lebih efektif dibandingkan metode lainnya
Keterkaitan	Penelitian yang akan dilakukan memiliki tujuan yang sama yaitu meningkatkan relevansi hasil pencarian namun objek dan metode yang digunakan berbeda

Dipublikasikan pada, Tahun	<i>Proceedings of the 27th annual international conference on Research and development in information retrieval – SIGIR '04, 2004</i>
Judul	<i>A STUDY OF METHODS FOR NORMALIZING USER RATINGS IN COLLABORATIVE FILTERING[1]</i>
Nama	<i>Jin, Rong; Si, Luo</i>
Tujuan	Tujuan dari collaborative filtering yaitu untuk membuat rekomendasi untuk pengguna tes dengan memanfaatkan rating informasi dari pengguna yang berbagi minat yang sama dengan pengguna tes. Dengan membandingkan dua strategi normalisasi yang berbeda untuk penilaian pengguna
Hasil	Dengan membandingkan dua metode normalisasi yang memanfaatkan rating informasi dari pengguna didapatkan bahwa metode normalisasi Decoupling lebih efektif dalam meningkatkan akurasi prediksi dibandingkan metode normalisasi Gaussian.
Keterkaitan	Penelitian yang akan dilakukan juga memanfaatkan rating untuk meningkatkan hasil pencarian

2.2 Dasar Teori

Pada sub-bab ini akan dijelaskan teori yang menjadi dasar penelitian ini, pertanyaan yang mendasari kajian pustaka adalah sebagai berikut:

1. Darimana sumber data yang akan didapatkan?
2. Bagaimana cara mendapatkannya?
3. Apakah relevansi hasil pencarian mempengaruhi pengalaman pengunjung?

4. Bagaimana cara agar dapat menampilkan produk yang sesuai dengan user?
5. Bagaimana cara agar produk yang dibutuhkan tampil menjadi rekomendasi yang paling atas?
6. Perbandingan Average Rating dibandingkan Wilson Score Interval

2.2.1 Google Play Store

Selain menyediakan semua fasilitas yang dimiliki oleh google, Google Playstore merupakan salah satu platform distribusi dari berbagai media dan aplikasi oleh Google yang paling sering digunakan oleh para pengguna android. Dengan didukung oleh Google Play Store, jutaan aplikasi dapat dengan mudah di download dan digunakan sesuai dengan kebutuhan dan keinginan pengguna. Sebelum menginstall aplikasi, play store akan menampilkan semua izin yang diperlukan seperti sebuah permainan perlu mengaktifkan getaran atau menyimpan data, mengaktifkan notifikasi, dan pengguna dapat memilih untuk menerima atau menolak izin tersebut. Google playstore disebut sebagai toko aplikasi dimana pengguna dapat mencari dan mendownload aplikasi mobile untuk perangkat android pilihan mereka. Aplikasi yang ada di dalam google playstore memiliki beragam karakteristik, apakah aplikasi tersebut gratis untuk di download atau pengguna perlu membayar terlebih dahulu. Aplikasi ini dapat langsung diunduh kedalam perangkat Android atau perangkat Google TV melalui aplikasi mobile google playstore, dan juga dapat digunakan melalui situs web google playstore untuk para pengguna android[6].

2.2.2 Web Scrapping

Web scrapping adalah proses pengambilan sebuah dokumen semi-terstruktur dari internet, umumnya berupa halaman

web dalam bahasa markup seperti HTML atau XHTML, dan menganalisis dokumen tersebut untuk diambil data tertentu dari halaman tersebut yang nantinya akan digunakan untuk kepentingan lain[9]. Web scrapping tidak dapat dimasukkan dalam sebuah bidang data mining karna data mining menyiratkan upaya untuk memahami pola semantik atau tren dari sejumlah besar data yang telah diperoleh.

Terdapat beberapa langkah dalam melakukan web scrapping, diantaranya adalah: [10]

1. Create Scrapping Template: Pembuat program mempelajari dokumen HTML dari website yang akan diambil informasinya untuk tag HTML yang memuat informasi yang akan diambil
2. Explore Site Navigation: Pembuat program mempelajari teknik navigasi pada website yang akan diambil informasinya untuk ditirukan pada aplikasi web scraper yang akan dibangun
3. Automate Navigation and Extraction: Aplikasi web scrapper dibuat untuk mengotomatisasi pengambilan informasi dari website yang ditentukan
4. Extracted Data and Package History: Informasi yang didapat akan disimpan dalam tabel database



Gambar 2.5. Visualisasi Web Scrapping [10].

Manfaat dari web scrapping adalah agar informasi yang didapatkan lenih terfokus sehingga memudahkan dalam melakukan pencarian sesuatu. Dengan memanfaatkan web scraping, mengumpulkan data akan jauh lebih mudah dilakukan, lebih cepat, dan juga lebih efisien.

2.2.3 Customer Relationship Management (CRM)

CRM atau Customer Relationship Management merupakan salah satu strategi dan usaha untuk menjalin hubungan dengan pelanggan dan memberikan pelayanan yang memuaskan bagi pelanggan. Dengan memanfaatkan CRM, maka perusahaan akan mengetahui apa yang diharapkan dan dibutuhkan oleh pelanggan. Dengan demikian maka kesetiaan pelanggan dapat dipertahankan dan pelanggan tidak akan berpindah kepada layanan yang lain. CRM mendukung suatu perusahaan untuk menyediakan pelayanan kepada pelanggan secara real time dengan menjalin hubungan dengan setiap pelanggan yang dimiliki. Website yang tidak dapat menampilkan apa yang dibutuhkan oleh pengguna akan ditinggalkan oleh penggunanya. Oleh karena itu dengan meningkatkan relevansi pada hasil pencarian pada prototipe website ini maka akan meningkatkan pengalaman pengguna sehingga membuat pengguna akan kembali menggunakan website ini. Dengan

meningkatkan kualitas pada prototipe website ini dengan menampilkan kebutuhan pengguna maka hal ini akan menjaga pengguna untuk tetap menggunakan website ini. Hal ini dapat membantu mempertahankan hubungan pelanggan dengan perusahaan[11].

Fase Customer Relationship Management

Menurut Kalakota dan Robinson [12], CRM terdiri dari tiga fase yaitu:

1. Mendapatkan pelanggan baru (Acquire)
2. Meningkatkan nilai pelanggan (Enhance)
3. Mempertahankan pelanggan yang telah ada (Retain)

Fitur search merupakan sumber terbesar untuk memudahkan akses informasi di internet. Untuk mendapatkan pelanggan baru maka sebuah website harus menyajikan konten yang tepat dalam waktu yang cepat sehingga website tersebut dapat meningkatkan jumlah pelanggan dan bersaing dengan website lainnya[13]. Dari ketiga fase yang disebutkan diatas prototipe website yang akan dibangun merujuk pada fase mendapatkan pelanggan baru (Acquire).

2.2.4 Teknologi Web

Web merupakan suatu sistem yang ada pada internet yang memungkinkan siapapun agar dapat menyediakan informasi. Web terdiri dari kumpulan dokumen elektronik di seluruh dunia. Setiap dokumen elektronik di web disebut webpage, yang dapat berisi teks, grafik, animasi, audio dan video[14]. Website merupakan suatu kebutuhan bagi masyarakat modern saat ini, baik itu digunakan untuk melakukan transaksi, penyebaran informasi, maupun pencarian informasi[15].

Seiring dengan berkembangnya teknologi web, para ahli melakukan klasifikasi atau memberikan penomoran untuk mengklasifikasikan generasi teknologi web yang digunakan. Klasifikasi tersebut terdiri dari[16].



Gambar 2.6. Evaluasi Web [16]

Web 1.0

Pada tahun 1989, Tim Burners-Lee menyarankan untuk membuat suatu ruang hypertext global untuk menciptakan suatu ruang informasi dimana orang dapat berkomunikasi dengan berbagi informasi[17]. Web 1.0 merupakan teknologi Web generasi pertama yang merupakan revolusi baru di dunia Internet karena telah mengubah cara kerja dunia industri dan media. Pada dasarnya, Website yang dibangun pada generasi pertama ini secara umum dikembangkan untuk pengaksesan informasi dan memiliki sifat yang sedikit interaktif.

Web 2.0

Istilah Web 2.0 pertama kalinya diperkenalkan oleh O'Reilly Media pada tahun 2004 sebagai teknologi Web generasi kedua yang mengedepankan kolaborasi dan sharing informasi secara online[17]. Menurut Tim O'Reilly, Web 2.0 dapat didefinisikan sebagai berikut: "Web 2.0 adalah revolusi bisnis di industri komputer yang disebabkan oleh penggunaan internet sebagai

platform, dan merupakan suatu percobaan untuk memahami berbagai aturan untuk mencapai keberhasilan pada platform baru tersebut. Salah satu aturan terutama adalah: Membangun aplikasi yang mengeksploitasi efek jaringan untuk mendapatkan lebih banyak lagi pengguna aplikasi tersebut”[16].

Pada umumnya, Website yang dibangun dengan menggunakan teknologi Web 2.0 memiliki fitur-fitur sebagai berikut:

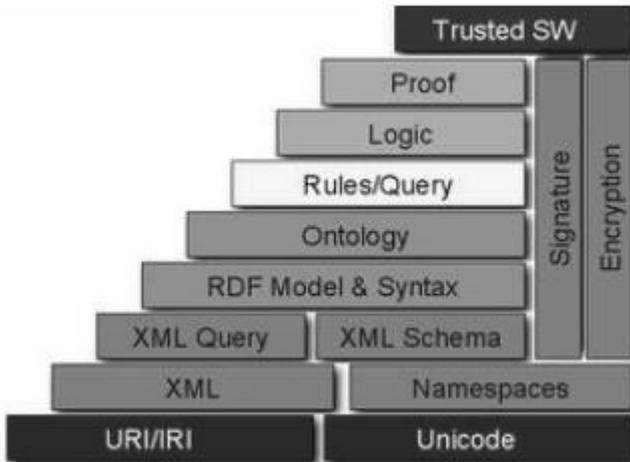
- CSS (Cascading Style Sheets)
- Aplikasi Rich Internet atau berbasis Ajax
- Markup XHTML
- Sindikasi dan agregasi data menggunakan RSS/Atom
- URL yang valid
- Folksonomies
- Aplikasi wiki pada sebagian atau seluruh Website
- XML Web-Service API

Web 3.0

Definisi untuk Web 3.0 sangat beragam mulai dari pengaksesan broadband secara mobile sampai kepada layanan Web berisikan perangkat lunak bersifat on-demand . Namun, menurut John Markoff, Web 3.0 adalah sekumpulan teknologi yang menawarkan cara baru yang efisien dalam membantu komputer mengorganisasi dan menarik kesimpulan dari data online[15].

Berdasarkan definisi yang dikemukakan tersebut, maka pada dasarnya Semantic Web memiliki tujuan yang sama karena Semantic Web memiliki isi Web yang tidak dapat hanya diekspresikan di dalam bahasa alami yang dimengerti manusia, tetapi juga di dalam bentuk yang dapat dimengerti, diinterpretasi dan digunakan oleh perangkat lunak (software agents). Melalui Semantic Web inilah, berbagai perangkat

lunak akan mampu mencari, membagi, dan mengintegrasikan informasi dengan cara yang lebih mudah[18].



Gambar 2.7. Layer Semantic Web

Web 4.0

Web 4.0 masih merupakan ide yang sedang berlangsung dan belum ada definisi atau gambaran yang jelas untuk mendeskripsikannya. Web 4.0 dikenal dengan web simbiosis, di mana interaksi antara manusia dan mesin dalam simbiosis. Hal ini memungkinkan untuk membangun suatu interface yang lebih kuat seperti pikiran yang mengontrol interface dengan menggunakan web 4.0. Sederhananya, mesin akan pintar dalam membaca konten suatu web, bereaksi dalam melakukan eksekusi dan memuruskan hal apa yang perlu di eksekusi pertama kali yang memuat situs cepat dengan kualitas terbaik dan performa dalam membangun suatu interface[16].

2.2.5 MySQL

MySQL adalah database server relasional yang gratis dibawah GNL (General Public License). Dengan sifatnya yang Open Source, memungkinkan juga user untuk melakukan modifikasi pada source code-nyaq untuk memenuhi kebutuhan spesifik mereka sendiri. MySQL merupakan database server multi-user dan multi-threaded yang tangguh (robust). Dengan memiliki banyak fitur MySQL bias bersaing dengan database komersial sekalipun[20]

2.2.6 Wilson Score Interval

Wilson Score Interval merupakan sebuah metode statistic yang diciptakan oleh Edwin B. Wilson pada tahun 1927 untuk menghitung confidence interval[3]. Berikut merupakan rumus dari Wilson Score Interval:

$$score = \frac{\hat{p} + \frac{z_{1-\alpha/2}^2}{2n} - z_{1-\alpha/2} \sqrt{\hat{p}(-\hat{p}) + \frac{z_{1-\alpha/2}^2}{4n}}}{1 + z_{1-\alpha/2}^2/n}$$

Dalam rumus di atas parameter didefinisikan dengan cara berikut:

- p adalah proporsi yang bernilai positif
- n adalah jumlah total penilaian
- $z_{\alpha / 2}$ adalah $(1-\alpha / 2)$ kuantil dari distribusi normal standar

2.2.7 Average Rating

Average Rating merupakan sebuah metode algoritma perhitungan untuk memberikan rating pada sebuah produk atau aplikasi[21]. Dengan perhitungan sebagai berikut:

$$Score = \frac{Positive\ Ratings}{Total\ Ratings}$$

2.2.8 Perbandingan Average Rating Dengan Wilson Score Interval

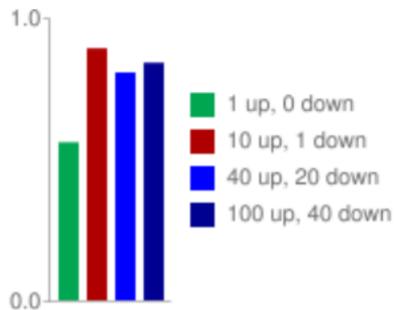
Average Rating bekerja dengan baik jika memiliki rating dalam jumlah besar, tetapi jika memiliki rating dalam jumlah kecil maka akan terjadi kekeliruan. Contoh item satu memiliki 2 positif rating dan 0 negatif rating. Item dua memiliki 100 positif rating dan 1 negatif rating. Maka dengan menggunakan Average Rating item dua (yang memiliki rating dalam jumlah besar) berada dibawah item satu (yang memiliki rating dalam jumlah kecil). Kesalahan seperti ini terdapat pada situs terkenal amazon.com[21].



Gambar 2.8. Penerapan Average Rating [21]

Kita perlu menyeimbangkan proporsi penilaian positif dengan ketidakpastian pengamatan dalam jumlah kecil. Wilson Score Interval memiliki keunggulan pada percobaan angka kecil dan probabilitas ekstrim. Randall mengatakan jika komen memiliki 1 upvote dan 0 downvote, ia memiliki tingkat upvote 100%, tapi karena tidak ada banyak data system akan menempatkan dekat dengan bagian bawah. Tetapi jika terdapat 10 upvotes dan 1 downvotes maka system memiliki keyakinan untuk menempatkan diatas dibandingkan dengan 40 upvotes dan 20 downvotes[22].

Berikut merupakan visualisasi voting menggunakan Wilson Score Interval.



Gambar 2.9. Visualisasi Wilson Score Interval [3]

Berikut merupakan table perbandingan Average Rating dengan Wilson Score Interval:

Tabel 2.2. Perbandingan Average Rating & Wilson Score Interval

Rangking	Average rating			Wilson Score Interval		
	Vote Up	Vote Down	Score	Vote Up	Vote Down	Score
Pertama	1	0	1	10	1	0,85
Kedua	10	1	0,91	100	40	0,83
Ketiga	100	40	0,71	40	20	0,80
Keempat	40	20	0,66	1	0	0,50

1. Merah 0,85 10 up 1 down
2. Biru tua 0,83 100 up 40 down
3. Biru muda 0,80 40 up 20 down

4. Hijau 0,50 1 up 0 down

Jika Average rating maka akan dimulai dari:

1. Hijau = $1 / 1 = 1$
2. Merah = $10 / 11 = 0,91$
3. Biru tua = $100 / 140 = 0,71$
4. Biru muda = $40 / 60 = 0,66$

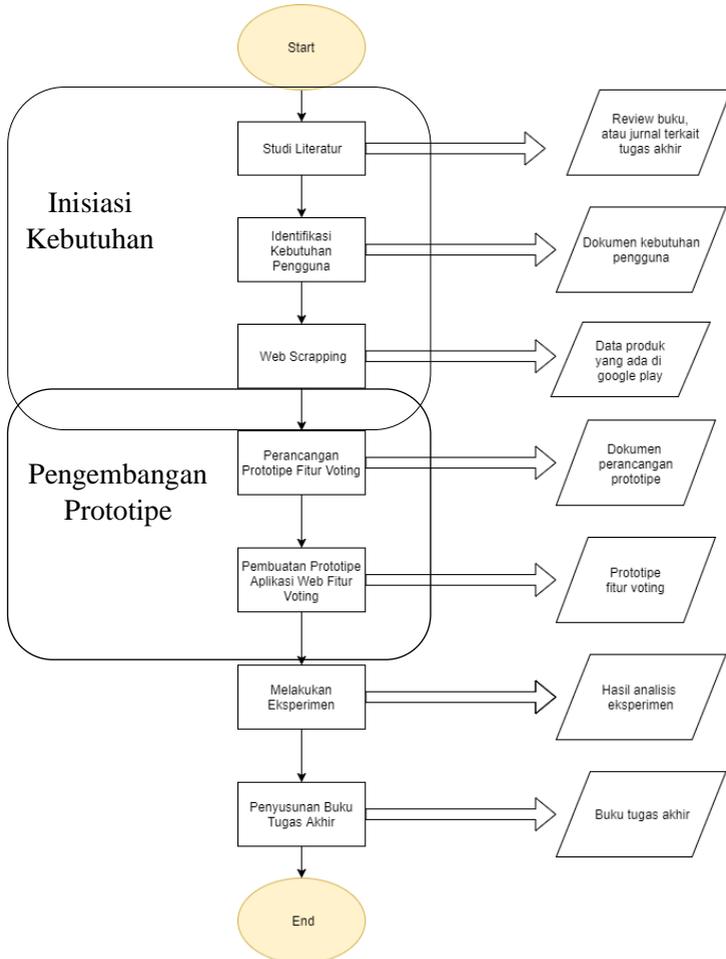
Dilihat dari tabel diatas peringkat yang dihasilkan Average Rating dengan Wilson Score Interval memiliki hasil yang berbeda. Pada Average Rating sebuah item yang memiliki 1 Vote Up dan 0 Vote Down akan berada pada posisi paling atas, sedangkan pada Wilson Score Interval berada diposisi paling bawah.

Dapat dilihat diatas bahwa Wilson Score Interval tidak mempedulikan berapa banyak vote yang dimiliki komentar, tetapi tentang berapa upvotes yang dibandingkan dengan jumlah total vote dan ukuran sampel.

BAB III METODOLOGI

Metodologi merupakan sebuah tahapan dalam penyelesaian permasalahan pada tugas akhir ini. Metodologi digunakan sebagai panduan pengerjaan tugas akhir.

3.1 Diagram Metodologi



Gambar 3.1. Metodologi penelitian

3.2 Uraian Metodologi Penelitian

3.2.1 Studi Literatur

Pada tahap studi literatur, penulis mengumpulkan informasi yang nantinya digunakan untuk mengusulkan solusi terkait dengan permasalahan pencarian dan penyaringan kebutuhan pengguna. Pengumpulan data dan informasi sendiri dilakukan dengan cara membaca referensi dari buku dan penelitian-penelitian yang telah dilakukan sebelumnya. Tujuan dari tahap ini agar penulis dapat memahami dasar teori yang berhubungan dengan permasalahan dan dapat mempermudah dalam menemukan solusi yang tepat.

3.2.2 Inisiasi Kebutuhan

Pada tahapan ini dilakukan inisiasi kebutuhan untuk mengetahui apa saja yang dibutuhkan untuk pembuatan aplikasi. Pada tahapan ini kebutuhan didapatkan melalui penelitian yang telah dilakukan sebelumnya dan melakukan wawancara kepada pengguna. Setelah mendapatkan kebutuhan dan mempelajari kajian, maka diperlukan pengambilan data Web Scraping dengan cara menggunakan Application Programming Interface (API). Data yang diperoleh akan disimpan dalam database.

3.2.2 Pengembangan Prototipe

Setelah dilakukan inisiasi kebutuhan dan pengumpulan data maka diketahui kebutuhan dari aplikasi, kemudian akan dilakukan perancangan dan pembuatan prototipe. Perancangan prototipe melakukan gambaran kasar mengenai aplikasi berupa User Interface. Pembuatan prototipe fitur voting berbasis website yang akan menggunakan framework Laravel PHP.

3.2.3 Penyusunan Buku Tugas Akhir

Pada tahapan ini akan dilakukan penyusunan laporan tugas akhir yang bertujuan sebagai pendokumentasian langkah-langkah pembuatan tugas akhir secara terperinci, hasil pembuatan tugas akhir, dan kesimpulan dari pengerjaan tugas akhir.

(halaman ini sengaja dikosongkan)

BAB IV

PERANCANGAN PENELITIAN

Bab ini akan menjelaskan mengenai perancangan tugas akhir dalam penelitian yang akan dilakukan sebagai panduan untuk melaksanakan penelitian tugas akhir yang terdiri dari perancangan pembuatan prototipe, rancang bangun prototipe, dan uji coba (*testing*).

4.1 Perancangan Pembuatan Protipe

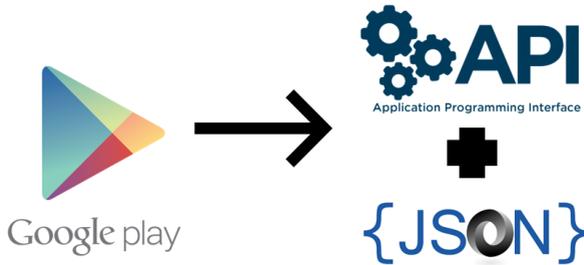
Bagian ini akan menjelaskan perancangan pembuatan prototipe sebagai acuan pada saat akan melakukan implementasi pada penelitian ini sehingga terjadinya kesalahan pada proses implementasi dapat diminimalisir.

4.1.1 Perancangan Pengambilan data

Pada tahapan ini akan dijelaskan langkah-langkah dalam melakukan pengambilan data yang bertujuan untuk pengolahan data mentah yang nantinya akan di implementasikan pada penelitian ini

4.1.1.1 Identifikasi Karakteristik Situs

Pada tahap ini akan dilakukan pengambilan data dari *Google Play Store*. Data yang diambil dari portal konten digital itu mengenai detail aplikasi yang ditampilkan disitus tersebut. Data yang diambil tidak langsung ke situs *Google Play Store*, melainkan dari API yang menggunakan JSON. API ini berasal dari `gooplayapi.herokuapp.com` yang dilakukan oleh Natanael Yabes W. Proses yang ada tersebut bisa dilihat pada gambar.



Gambar 4. 1. Alur Pengambilan Data Dari API JSON

Dalam melakukan pengambilan data menggunakan konsep API dilakukan secara *real time* ke situs portal konten digital. Pengambilan data yang dilakukan melalui *curl client* yang menggunakan JSON yang sudah disediakan pada *google play store* dan kemudian dimasukkan kedalam *database google play store*. Data yang dimasukkan ke database ini bersifat general dan spesifik. Data general ini adalah data yang memiliki banyak aplikasi tetapi hanya menampilkan data secara umum terkait aplikasi tersebut. Data spesifik adalah data yang berisikan data satu aplikasi namun ditampilkan secara detail. Data general dan spesifik ini dijadikan satu tabel aplikasi pada *database* dengan atribut yang dapat melihat pada table yang memiliki alur proses pengambilan data dari API JSON seperti gambar berikut:



Gambar 4. 2. Alur Pengambilan Data Dari Proses API JSON

Data yang dibutuhkan oleh penelitian ini berupa data-data produk aplikasi. Pengambilan data pada Google Play API dilakukan secara *real time* dengan struktur seperti tabel dibawah ini:

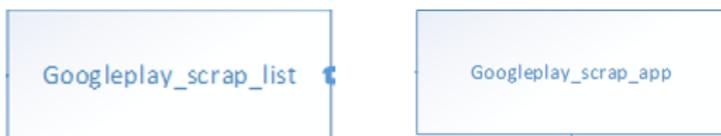
Tabel 4. 1. Deskripsi Data Atribut Pada Tabel Applications

Nama Atribut	Tipe Data	Keterangan
Id(PK)	Int(11)	<i>Primary Key</i> untuk tabel applications
title	Varchar(100)	Nama dari aplikasi
summary	Varchar(100)	outline/abstrak/tag dari aplikasi
icon	Varchar(500)	gambar/logo/symbol dari aplikasi
price	Varchar(1)	harga jual dari aplikasi
free	Tinyint(11)	aplikasi tidak memiliki harga jual (true/false)
minInstalls	Int(11)	installasi minimal dari aplikasi
maxInstalls	Int(11)	installasi maksimal dari aplikasi
score	Decimal(10,2)	rata-rata skor rating dari aplikasi
reviews	Int(11)	jumlah review yang diberikan untuk aplikasi
developer	Varchar(45)	nama dari pengembang aplikasi
developerEmail	Varchar(45)	alamat email milik pengembang aplikasi

developerWebsite	Varchar(45)	alamat situs milik pengembang aplikasi
update	Varchar(45)	tanggal terakhir aplikasi diperbarui
genre	Varchar(45)	kategori dari aplikasi
genreId	Varchar(45)	id pada googleplay untuk kategori aplikasi
description	text	deskripsi/gambaran menyeluruh dari aplikasi
descriptionHTML	text	deskripsi/gambaran menyeluruh dari aplikasi versi HTML
offersIAP	Tinyint(1)	aplikasi menyediakan pembelian di dalam aplikasi (In App Purchase) (true/false)
adSupported	Tinyint(1)	terdapat iklan dalam aplikasi
androidVersionText	Varchar(45)	versi android yang dibutuhkan agar aplikasi dapat diinstall dalam bentuk text pada googleplay
androidVersion	Varchar(10)	versi android yang dibutuhkan agar

		aplikasi dapat diinstall
contentRating	Varchar(10)	rating usia untuk pengguna dari aplikasi
video	Varchar(100)	link video dari aplikasi
recentChanges	Varchar(45)	gambaran perubahan yang terjadi pada versi aplikasi saat ini
preregister	Tinyint(1)	aplikasi dapat dicoba terlebih dahulu (true/false)
url	Varchar(100)	link aplikasi dapat diunduh pada googleplay
appId	Varchar(100)	id aplikasi yang diberikan oleh googleplay untuk aplikasi

Tabel diatas adalah hasil dari proses pengambilan data yang dilakukan kemudian hasil tersebut diletakkan pada penyimpanan sementara di PHP. Kemudian data yang tersimpan tadi dimasukan kedalam MySQL dan terdapat 100 data aplikasi Play Store yang diperoleh.



Gambar 4. 3. Entitas Play Store

4.1.2 Desain Basis Data

Pada tahapan ini berisikan desain database yang bertujuan dalam perancangan database yang terstruktur dengan membuat relasi antar tabel yang pada pengerjaanya memiliki berbagai tahapan dimana kolom-kolom pada tabel tersebut tidak berisikan data yang redundan.

4.1.2.1 Perancangan Basis Data Konseptual

Perancangan basis data konseptual merupakan proses membuat suatu model informasi yang digunakan pada suatu perusahaan, yang terlepas dari semua pertimbangan fisik. Tahapan didalam perancangan basis data konseptual yaitu membangun model data konseptual untuk setiap tampilan user.

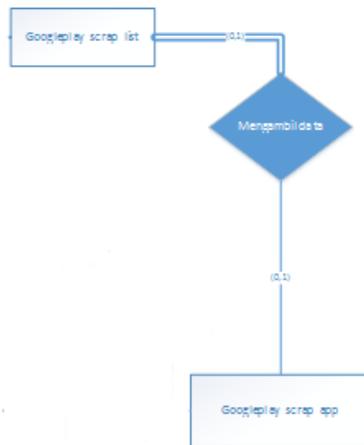
4.1.2.1.1 Identifikasi Tipe Entity

Pada bagian ini bertujuan untuk mengidentifikasi tipe utama yang dibutuhkan dari sudut pandang tertentu. Salah satu cara yang digunakan dengan mengidentifikasi kata benda utama dan objek-objek yang berkaitan dengan spesifikasi.

Sehingga untuk mencakup play store yang memiliki sifat general dan spesifik, maka dibuat tabel list untuk bagian general dan table app untuk bagian spesifik.

4.1.2.1.2 Identifikasi Tipe Relasi

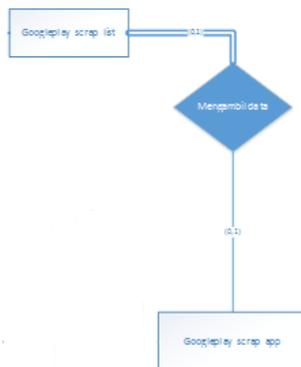
Pada bagian ini bertujuan untuk mengidentifikasi relasi penting yang ada antara berbagai tipe entity yang telah diidentifikasi. Salah satu metode yang digunakan merupakan mempelajari kata kerja yang berkaitan dengan entity, tipe relasi, dan spesifikasi.



Gambar 4. 4. Relasi Entitas

4.1.2.1.3 Identifikasi Dan Menghubungkan Atribut Dengan Entiti Ataupun Dengan Tipe Relasi

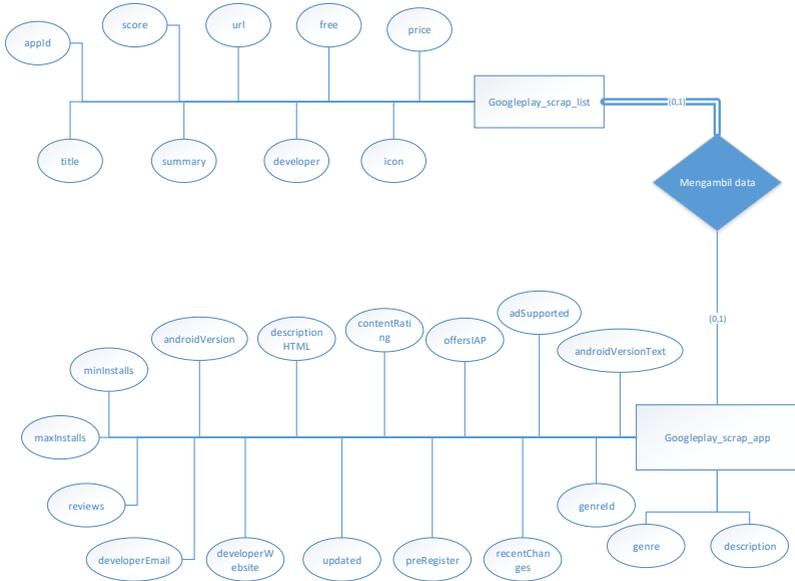
Pada bagian bertujuan memiliki tujuan untuk menghubungkan atribut dengan tipe entity dan relasi yang sesuai. Dalam menentukan atribut harus mampu mengidentifikasi simple atau composite atribut, single atau multi-valued atribut dan atribut turunan.



Gambar 4. 5. Relasi Antar Entitas

4.1.2.1.4 Menentukan Domain Atribut

Pada tahapan ini memiliki tujuan untuk menentukan wewenang bagi atribut dalam model data konseptual. Domain adalah satu kumpulan nilai dimana menggambarkan nilai-nilai dari satu atau lebih atribut.



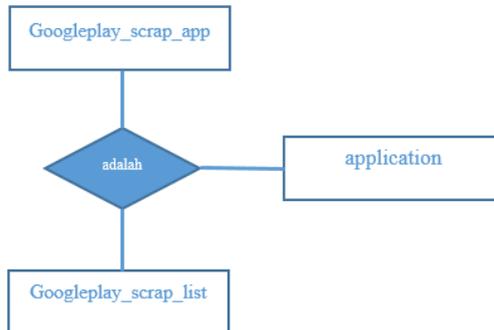
Gambar 4. 6. Menentukan Domain Atribut

4.1.2.1.5 Menentukan Atribut Candidate Key Dan Primary

Tahap selanjutnya bertujuan untuk mengidentifikasi candidate key dari setiap relasi dan jika ditemukan lebih dari satu candidate key pada satu relasi maka akan dipilih salah satu untuk menjadi primary key. Primary key pada Play Store yaitu Id sedangkan appId merupakan candidate key.

4.1.2.1.6 Mempertimbangkan Penggunaan Konsep Pemodelan Yang Lebih Baik

Pada tahapan ini memiliki tujuan untuk mempertimbangkan konsep pemodelan yang lebih baik seperti spesialisasi, generalisasi, agregasi, atau komposisi. Namun pada tahapan ini, yang terjadi hanyalah proses generalisasi, dimana terjadi proses peleburan di tabel `googleplay_scrap_app` dan `googleplay_scrap_list` menjadi tabel `applications`. Ini dikarenakan atribut yang ada di `googleplay_scrap_list` memiliki kesamaan terhadap `googleplay_scrap_app`. Demi menghemat memori dan menghilangkan redundansi, maka dibutuhkan proses generalisasi.



Gambar 4. 7. Proses Generalisasi Pada Playstore

4.1.2.1.7 Memeriksa Dengan Model Redundansi

Pada tahapan ini memiliki tujuan untuk memeriksa keberadaan redundansi dalam model basis data. Aktivitas yang dilakukan pada tahapan ini adalah menentukan ulang relasi 1:1 dan membuang relasi yang redundan. Namun pada tahapan ini, tidak ada perubahan pada basis data konseptual dikarenakan kondisi tabel yang ada tidak memerlukan perlakuan 2 aktivitas pada tahapan ini

4.1.2.1.8 Data Dictionary

Data dictionary bisa juga disebut *system catalog* dan bisa diartikan menjadi hasil kompilasi *data definition language* yang kumpulan tabelnya tersimpan secara kolektif.

Biasanya, di dalam sebuah *data dictionary* terdapat beberapa hal berikut:

- Definisi dari semua objek skema yang terdapat di dalam sebuah database (tabel, view, indeks, cluster, sinonim, urutan, prosedur, dll)
- Berapa banyak alokasi data yang digunakan oleh objek skema tersebut
- Nilai default untuk setiap kolom
- Nama dari pengguna
- Keuntungan dan peran dari setiap pengguna yang telah diperbolehkan
- Informasi yang telah diaudit, seperti siapa yang dapat mengakses atau memperbaharui objek skema tersebut
- Dan informasi umum mengenai basis data lainnya

Data dictionary biasanya mempunyai tiga pengguna utama, yaitu :

- Administrator yang dapat mengakses *data dictionary* tersebut untuk mengetahui informasi mengenai pengguna, objek skema dan struktur penyimpanan dari basis data
- Administrator yang dapat memodifikasi *data dictionary* setiap kali ada DDL (*Data Definition Language*) yang ditambahkan
- Pengguna lainnya yang dapat menggunakan *data dictionary* sebagai referensi yang dapat dibaca untuk mengetahui informasi apapun mengenai basis data

Pembuatan *data dictionary* berguna untuk mengetahui jenis tipe apa yang akan digunakan pada setiap kolom data pada basis data

yang digunakan dan merupakan penjelasan lebih rinci dari ERD (*Entity Relation Diagram*) dari sebuah basis data. Pada penelitian kali ini, akan ada dua tabel yang akan digunakan yaitu tabel `playapplication` untuk menyimpan data play store dan tabel `appstoreapplication` untuk menyimpan data dari app store. *Data Dictionary* pada tabel `playapplication` dapat dilihat pada tabel berikut.

Tabel 4. 2. Data Dictionaru Pada Applications

Table	Column	Xtype	Data Type	Lenght	Null	Description
applications	id	PK, FK	int	11	N	<i>Primary Key</i> untuk tabel applications
applications	title		varchar	100	N	Nama dari aplikasi
applications	summary		varchar	100	N	outline/abstrak/tag dari aplikasi
applications	icon		varchar	500	N	gambar/logo/symbol dari aplikasi
applications	price		varchar	1	N	harga jual dari aplikasi
applications	free		tinyint	11	N	aplikasi tidak memiliki harga jual (true/false)
applications	minInstalls		int	11	N	instalasi minimal dari aplikasi
applications	maxInstalls		int	10,2	N	instalasi maksimal dari aplikasi
applications	score		decimal	11	N	rata-rata skor rating dari aplikasi

applications	reviews		int	42	N	jumlah review yang diberikan untuk aplikasi
applications	developer		varchar	45	N	nama dari pengembang aplikasi
applications	developerEmail		varchar	45	N	alamat email milik pengembang aplikasi
applications	developerWebsite		varchar	45	N	alamat situs milik pengembang aplikasi
applications	update		varchar	45	N	tanggal terakhir aplikasi diperbarui
applications	genre		varchar	45	N	kategori dari aplikasi
applications	genreId		varchar	45	N	id pada googleplay untuk kategori aplikasi
applications	description		text		N	deskripsi/gambaran menyeluruh dari aplikasi
applications	descriptionHTML		text		N	deskripsi/gambaran menyeluruh dari aplikasi versi HTML

applications	offersIAP		tinyint	1	N	aplikasi menyediakan pembelian di dalam aplikasi (In App Purchase) (true/false)
applications	adSupported		tinyint	1	N	terdapat iklan dalam aplikasi
applications	androidVersionText		varchar	45	N	versi android yang dibutuhkan agar aplikasi dapat diinstall dalam bentuk text pada googleplay
applications	androidVersion		varchar	10	N	versi android yang dibutuhkan agar aplikasi dapat diinstall
applications	contentRating		varchar	10	N	rating usia untuk pengguna dari aplikasi
applications	video		varchar	100	N	link video dari aplikasi
applications	recentChanges		varchar	45	N	gambaran perubahan yang terjadi pada versi aplikasi saat ini

applications	preregister		tinyint	1	N	aplikasi dapat dicoba terlebih dahulu (true/false)
applications	url		varchar	100	N	link aplikasi dapat diunduh pada googleplay
applications	appld		varchar	100	N	id aplikasi yang diberikan oleh googleplay untuk aplikasi

4.1.2.2 Perancangan Basis Data Logikal

Perancangan basis data logical adalah proses membangun suatu model informasi yang digunakan yang digunakan didalam perusahaan berdasarkan model data tersebut, tanpa memperhatikan pertimbangan penggunaan suatu DBMS dan perangkat keras lainnya. Perancangan ini bertujuan untuk membangun model data logikal yang mengacu pada model data logikal dan juga memvalidasi model agar mendukung transaksi yang dibutuhkan oleh user.

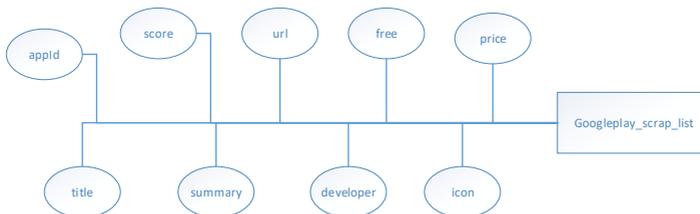
4.1.2.2.1 Menentukan Relasi Untuk Model Data Logikal

Pada tahapan ini bertujuan untuk membuat suatu relasi untuk model data logikal yang menggambarkan entity, relasi, dan atribut yang telah di identifikasi. Hasil dari tahapan ini terbagi menjadi sembilan proses dalam menentukan relasi model data.

4.1.2.2.1.1 Mapping of Strong Entity Types

Mapping of strong entity types dilakukan untuk memetakan primary key pada entitas utama.

- Entitas `Googleplay_scrap_list`



Gambar 4. 8 Entitas `Googleplay_scrap_list`

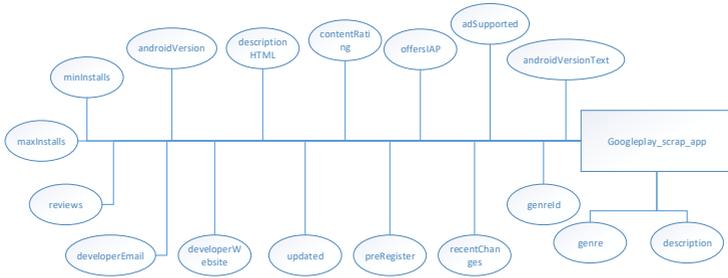
Entitas `Googleplay_scrap_list` dalam MRD adalah sebagai berikut:

googleplay_scrap_list

appId	title	summary	icon	price	free	score	developer	url
-------	-------	---------	------	-------	------	-------	-----------	-----

Gambar 4.9 MRD Googleplay_scrap_list

- Entitas Googleplay_scrap_app



Gambar 4.10 Entitas Googleplay_scrap_app

Entitas Googleplay_scrap_app dalam MRD adalah sebagai berikut:

googleplay_scrap_app

description	androidVersion	mininstalls	maxinstalls	description HTML	contentRating	offersIAP	adSupported	androidVersionText	reviews
-------------	----------------	-------------	-------------	------------------	---------------	-----------	-------------	--------------------	---------

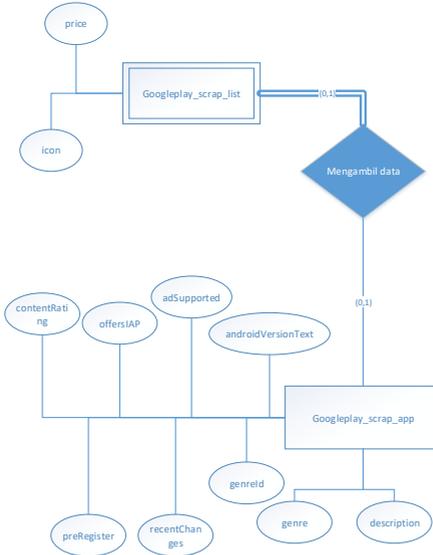
developerEmail	genreId	developerWebsite	updated	preRegister	recentChanges	genre	appId
----------------	---------	------------------	---------	-------------	---------------	-------	-------

Gambar 4.11 MRD Googleplay_scrap_app

4.1.2.2.1.2 Mapping of Weak Entity Types

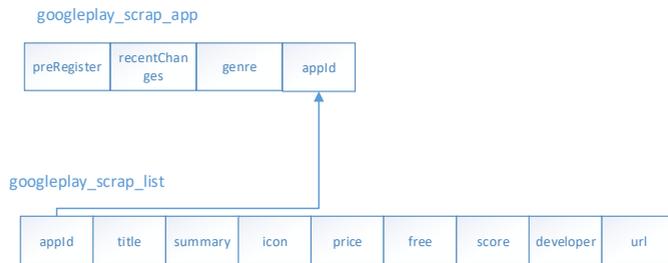
Mapping of weak entity types dilakukan untuk memetakan entitas yang tidak memiliki primary key kepada entitas terkait.

- Entitas Googleplay_scrap_app



Gambar 4.12 Mapping Weak Entity

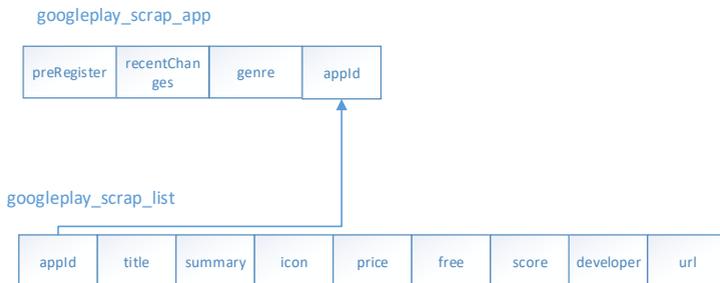
Weak Entitas Googleplay_scrap_app yang berelasi dengan entitas Googleplay_scrap_list dalam MRD adalah sebagai berikut:



Gambar 4.13 MRD Mapping Weak Entity

4.1.2.2.1.3 Mapping of Binary (1:1) Relationship Types

Entitas `googleplay_scrap_list` yang berelasi one to one dengan entitas `googleplay_scrap_app` dalam MRD adalah sebagai berikut:



Gambar 4. 14 Mapping Binary (1:1)

4.1.2.2.1.4 Mapping of Binary (1:N) Relationship Types

Tidak dilakukan karena dalam ERD tidak ada atribut yang binary 1:N relationship types.

4.1.2.2.1.5 Mapping of Binary (M:N) Relationship Types

Tidak dilakukan karena dalam ERD tidak ada atribut yang binary M:N relationship types

4.1.2.2.1.6 Mapping of Multivalued Attributes

Tidak dilakukan karena dalam ERD tidak ada atribut yang *multivalued*.

4.1.2.2.1.7 Mapping of N-ary Relationship Types

Tidak dilakukan karena dalam ERD tidak ada atribut yang N-Ary Relationship Type.

4.1.2.2.1.8 Mapping of Specialization or Generalization

Tidak dilakukan karena dalam ERD tidak ada spesialisasi atau generalisasi.

4.1.2.3 Perancangan Basis Data Fisikal

Perancangan basis data fisikal adalah proses pembuatan deskripsi implementasi dari basis data media *secondary storage*. Perancangan basis data fisikal menggambarkan hubungan dasar organisasi file pengindeksan yang digunakan untuk memperoleh akses yang efisien pada data, serta Batasan integritas dan ukuran yang berhubungan. Langkah-langkah dalam metodologi perancangan basis data fisikal adalah:

4.1.2.3.2 Merancang Relasi Dasar

Pada tahapan ini, dalam mentranslasi model data logikal pada bagian relasi, diperlukan penyesuaian pada atribut dan relasinya. Dalam hal ini, atribut yang ada bisa dilihat di data dictionary, lalu untuk relasi bisa dilihat pada gambar berikut:



Gambar 4. 15. Relasi Applications dan Tes

Relasi yang ada yaitu app_id dimana primary key berada di id table applications dan id pada tabel *tes*.

4.1.2.3.1 Menerjemahkan Model Data Logikal Ke Target DBMS

Pada tahapan ini bertujuan untuk menghasilkan skema basis data relasional dari model data logikal yang dapat diimplementasikan kedalam suatu DBMS. Aktivitas yang dilakukan meliputi merancang relasi dasar, merancang representasi dari data turunan, dan merancang *general constraints*.

4.2 Rancang Bangun Prototipe

Hal pertama yang akan dilakukan sesuai dengan tahapan ICONIX adalah membuat user interface kemudian membuat use case diagram. Selanjutnya dari use case diagram yang telah dibuat akan diturunkan menjadi class diagram. Kebutuhan dari perancangan prototipe didapatkan dari proses pencarian kebutuhan.

4.2.1 User Interface

User interface dibuat untuk mempermudah pengguna dalam menggunakan aplikasi yang dibuat. Gambar dibawah adalah *user interface* dari halaman fitur voting.

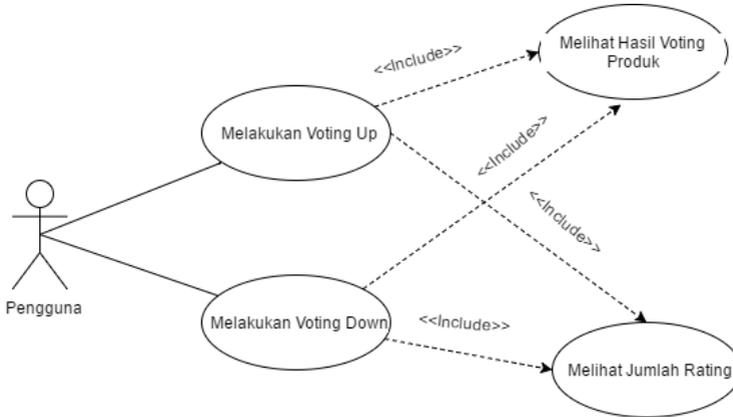
Prototipe Fitur Voting					
Title	Rating	RatingUps	RatingDowns	Id	Action
Spotify	0,00000	0	0	01	Up Down
Shazam	0,00000	0	0	02	Up Down
Soundcloud	0,00000	0	0	03	Up Down
Smule	0,00000	0	0	04	Up Down
Samsung	0,00000	0	0	05	Up Down
Soundhound	0,00000	0	0	06	Up Down
Google Play	0,00000	0	0	07	Up Down
Music	0,00000	0	0	08	Up Down

Gambar 4.16 User Interface Prototipe

Halaman ini berisikan nama aplikasi, hasil perhitungan *rating*, jumlah *rating up* dan *down*, id aplikasi, dan *button up* dan *down* untuk melakukan *voting* aplikasi. Halaman ini akan menampilkan jumlah *rating up* dan *down* setelah *user* melakukan *voting*, dan kemudian akan menampilkan hasil *rating* aplikasi yang akan mengurutkan aplikasi dari hasil *rating* tertinggi hingga terendah.

4.2.2 Use Case Diagram

Use Case dibuat untuk mendetailkan aktivitas yang akan dapat dilakukan oleh pengguna. Berikut ini merupakan use case dari prototipe yang dirancang.



Gambar 4. 17. Use Case Diagram

4.2.2.1 Daftar Use Case

Berikut ini merupakan daftar *use case* dari diagram yang telah dirancang:

Tabel 4. 3. Daftar Use Case

Kode	Aktor	Test Case
UCP1	Pegguna	Melakukan Voting Up
UCP2	Pegguna	Melakukan Voting Down
UCP3	Pegguna	Melihat Jumlah Rating
UCP4	Pegguna	Melihat Hasil Voting Produk

4.2.3 Descriptive Case

Berikut ini merupakan alur yang menjelaskan use case poin dari setiap aktivitas yang ada pada use case diagram.

Tabel 4. 4. Use Case Point 1.

UCP-1: Melakukan Voting Up
Normal Flow: Pengguna melakukan voting up dengan mengklik tombol “up” yang ada pada aplikasi
Alternate Flow: -

Tabel 4. 5. Use Case Point 2

UCP-2: Melakukan Voting Down
Normal Flow: Pengguna melakukan voting down dengan mengklik tombol “down” yang ada pada aplikasi
Alternate Flow: -

Tabel 4. 6. Use Case Point 3

UCP-3: Melihat Jumlah Rating
Normal Flow: Pengguna melakukan voting up atau down pada aplikasi dan kemudian jumlah rating dapat dilihat setelah pengguna melakukan voting
Alternate Flow: Pengguna dapat melihat jumlah rating berdasarkan hasil voting sebelumnya

Tabel 4.7 Use Case Point 4

UCP-4: Melihat Hasil Voting Produk
Normal Flow: Pengguna melakukan voting up atau down kemudian aplikasi akan menampilkan hasil perhitungan produk berdasarkan input voting dari pengguna
Alternate Flow: Pengguna dapat melihat jumlah rating berdasarkan hasil voting sebelumnya

(halaman ini sengaja dikosongkan)

BAB V IMPLEMENTASI

Pada bab implementasi ini dijelaskan hasil implementasi yang telah dilakukan dalam tugas akhir terkait pengembangan aplikasi pada tiap aspek diteliti, yakni pencarian kebutuhan, pengambilan data, rancang bangun prototipe, dan validasi prototipe.

5.1 Lingkungan Implementasi

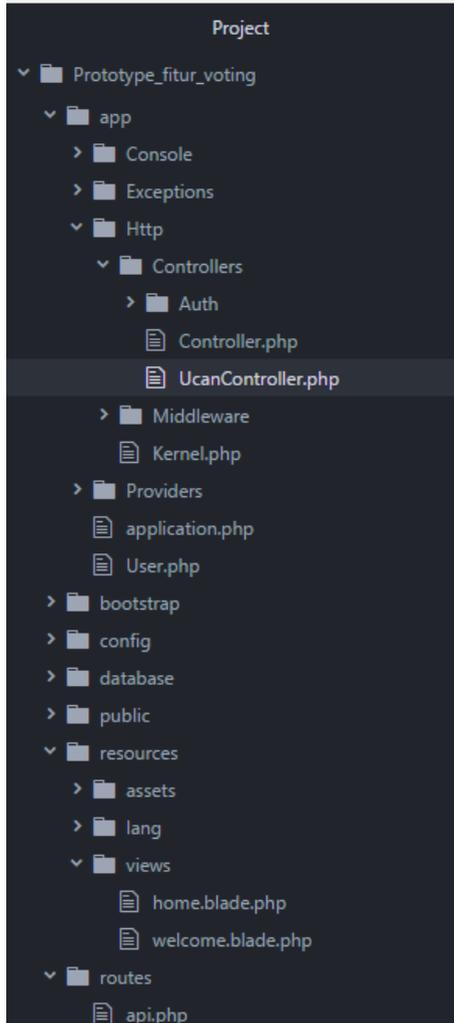
Pada pengembangan aplikasi, peneliti menggunakan perangkat keras dengan spesifikasi seperti pada Tabel... Sedangkan untuk perangkat lunak yang digunakan dalam pengembangan aplikasi adalah seperti pada Tabel 5.1

Tabel 5. 1. Spesifikasi Perangkat Lunak

Perangkat keras	Spesifikasi
Laptop	Prosesor: Inter(R) Core (TM) i-7-3537u CPU @ 2.00GHz (4 CPUs), ~2.5GHz
	RAM: 4GB
	Sistem Operasi Windows 10
Perangkat lunak	Spesifikasi
Web server	Apache
Database server	MySQL
Bahasa	PHP
Framework	Laravel
Editor	Atom

5.2 Struktur Direktori

Berikut adalah struktur direktori untuk pengembangan aplikasi pencarian dan penyaringan produk aplikasi pada Google Play.



Gambar 5. 1. Struktur Direktori

5.3 Pengambilan Data

Setelah menentukan data yang perlu diambil, maka selanjutnya data diambil menggunakan Bahasa pemrograman PHP dari Google Play API. Pengambilan data dilakukan dengan *Web Scrapping* dengan menggunakan CURL dimana data yang akan diambil berupa JSON. JSON yang berhasil diambil akan dilakukan *decode* untuk mendapatkan nilai yang ada di dalamnya, dibutuhkan wadah untuk menempatkan nilai JSON yaitu variabel. Sebelum ditempatkan di variabel, perlu diketahui nilai yang dibawa JSON. Variabel yang sudah membawa nilai ditempatkan didalam kode yang berfungsi untuk memasukkan data ke dalam basis data.

5.3.1 Pembuatan Scrapper

Pada tahapan ini akan menjelaskan bagaimana proses scrapping dilakukan yang akan mengambil data list dan data app. List sendiri yaitu proses mengambil daftar aplikasi dari suatu koleksi. App adalah proses mengambil data aplikasi secara utuh dari aplikasi. List dan app akan mengambil data di Google Play API.

Tabel 5.2 Kode Untuk Konfigurasi Koneksi Database Googleplayapi

Koneksi.php
<pre> <?php \$db_user = 'root'; \$db_pass = ""; \$db_name = 'googleplayapi'; \$dsn = 'mysql:dbname=' . \$db_name . ';charset=utf8;host=localhost'; try { \$dbh = new PDO(\$dsn, \$db_user, \$db_pass); } catch (PDOException \$e) { echo 'Koneksi Gagal : ' . \$e->getMessage(); </pre>

```

}
?>

```

Kode pada gambar 5.2 merupakan konfigurasi untuk koneksi database googleplayapi.

Tabel 5.3 Kode Untuk Lanjutan Proses Pengambilan Data

Scrapp_list.php
<pre> \$service_url = "https://gooplayapi.herokuapp.com/list/top_free/music_and _audio/110/en/en/100/199"; \$curl = curl_init(\$service_url); curl_setopt(\$curl, CURLOPT_RETURNTRANSFER, 1); curl_setopt(\$curl, CURLOPT_SSL_VERIFYHOST, 0); curl_setopt(\$curl, CURLOPT_SSL_VERIFYPEER, 0); //execute the session \$curl_response = curl_exec(\$curl); //finish off the session curl_close(\$curl); \$curl_jason = json_decode(\$curl_response, true); </pre>

Kode pada Tabel 5.3 merupakan lanjutan dari proses pengambilan data kategori. Kategori yang ada pada Google Play API dimasukkan ke proses CURL (*client URL*) untuk mendapatkan data berbentuk json. Data json yang didapat kemudian di *decode*. Untuk mengeluarkan nilai dari json yang telah di *decode* perlu dilakukan *looping*.

Tabel 5.4 Kode Untuk Proses Pengelolaan Data

```

Scrapp_list.php
    foreach ($curl_jason as $val) {
        $url = $val['url'];
        $title = $val['title'];
        $appid = $val['appId'];
        $summary = $val['summary'];
        $developer = $val['developer'];
        $icon = $val['icon'];
        if(isset($val['score'])){
            $score =
            htmlspecialchars($val['score'], ENT_QUOTES);
        }
        else{
            $score = "";
        }
        $score =
        htmlspecialchars($val['score'],
        ENT_QUOTES);
    }
    else{
        $score = "";
    }

    $price = $val['price'];
    $free = $val['free'];

```

Kode pada Tabel 5.4 merupakan proses pengolahan data yang diambil dari cURL (*client* URL). Agar data json yang didapat tidak hanya berwujud data mentah, dilakukan konversi karakter data json yang belum terdefinisi menjadi entitas HTML dengan `htmlspecialchars`.

Tabel 5.5 Kode Untuk Melakukan Input Data

```

Scrapp_list.php
$query = "select appId from applications where
appId='".$appid.'";
$result1 = $dbh->query($query)->fetch();
if($result1==false){
    $table = 'applications';
    $field =
    `url`,`title`,`appId`,`summary`,`developer`,`icon`,`score`,`pr
ice`,`free`; // kolomnya, kalo > 1 pisahkan pakai koma
    $val = '?,?,?,?,?,?,?,?';
    $array = array( $url, $title, $appid, $summary,
    $developer, $icon, $score, $price, $free ); // sesuai jumlah
    kolom juga
    $sth = $dbh->prepare( "INSERT INTO $table ($field)
VALUES ($val) " );
    $input = $sth->execute( $array );
}
}

```

Kode pada Tabel 5.5 merupakan proses memasukan data dengan pengecekan kondisi, yaitu aplikasi yang sudah ada pada basis data tidak akan masuk lagi ke dalam basis data. Cara untuk mengeceknya adalah dengan mencocokkan appId. appId ini nantinya akan dijadikan sebagai *unique key*.

Lalu proses selanjutnya, yaitu mengambil data spesifik aplikasi secara utuh. Proses yang membedakan antara list dengan app adalah pemanggil untuk *looping*-nya tidak melalui kategori seperti hal-nya pada list, melainkan pada appId yang berfungsi mengambil data sisa yang belum terambil dari proses list. Lalu pembeda lainnya berada di data yang bersifat array, diperlukan pemilahan data. Berikut adalah potongan kode pengambilan

data spesifik aplikasi di Google Play yang memiliki ekstensi php.

Tabel 5.6 Kode Untuk Proses Pengambilan Data

<pre>Scrapp_app.php \$query = "select appid from applications"; \$result1 = \$dbh->query(\$query)->fetchAll(); foreach(\$result1 as \$r1){ \$a = \$r1['appid'];}</pre>
--

Kode pada Gambar __ adalah proses pengambilan data yang belum terambil dari proses list dengan mengacu pada appId produk aplikasi

Tabel 5.7 Kode Untuk Proses Konversi Data

<pre>Scrapp_app.php if(isset(\$vali['minInstalls'])){ \$minInstalls = htmlspecialchars(\$vali['minInstalls'], ENT_QUOTES); } else{ \$minInstalls = ""; } // \$minInstalls = htmlspecialchars(\$vali['minInstalls'], ENT_QUOTES); if(isset(\$vali['maxInstalls'])){ \$maxInstalls = htmlspecialchars(\$vali['maxInstalls'], ENT_QUOTES); } else{ \$maxInstalls = "";</pre>
--

```

    }
    // $maxInstalls = htmlspecialchars($vali['maxInstalls'],
    ENT_QUOTES);
    if(isset($vali['reviews'])){
        $reviews = htmlspecialchars($vali['reviews'],
    ENT_QUOTES);
    }

```

Kode pada Tabel 5.7 merupakan proses mengkonversi data yang didapat menjadi entitas HTML dengan `htmlspecialchars` dan memilah data ke array.

Tabel 5.8 Kode Untuk Proses Pemasukan Data Hasil Konversi

Scrapp_app.php

```

$query2 = "UPDATE applications SET minInstalls = '$minInstalls', maxInstalls = '$maxInstalls', reviews = '$reviews', developerEmail = '$developerEmail', developerWebsite = '$developerWebsite', updated = '$updated', genre = '$genre', genreId = '$genreId', description = '$description', descriptionHTML = '$descriptionHTML', offersIAP = '$offersIAP', video = '$video', adSupported = '$adSupported', androidVersion = '$androidVersion', contentRating = '$contentRating', androidVersionText = '$androidVersionText', recentChanges = '$recentChanges', preregister = '$preregister' WHERE appId = '$a'";

if($dbh->exec($query2))
{
}
else {
    echo 'gagal';
    echo '<a href="index.php">Kembali</a>';
}

```

Kode pada Tabel 5.8 diatas merupakan proses sistem dalam memasukan data yang didapat dan terkonversi dari pengambilan app pada Tabel 5.7 dan Tabel 5.6 ke dalam basis data.

5.4 Rancang Bangun Prototipe

Tahapan ini merupakan implementasi dari tahapan analisis dan perancangan ke dalam bentuk aplikasi sesuai dengan penjelasan lingkungan implementasi.

5.4.1 Implementasi Halaman Prototipe Fitur Voting

Halaman Fitur Voting menampilkan seratus aplikasi yang telah diambil dari GooglePlay menggunakan Scrapping. Berikut merupakan kode yang berada pada direktori Views pada framework Laravel.

Tabel 5.9 Kode Untuk Menampilkan Nama Tabel Pada Halaman Fitur Voting

<pre> home.blade.php <body> <div id="isitabel">
 <table id="tabelku"> <thead> <th>title</th> <th>rating</th> <th>ratingups</th> <th>ratingdowns</th> <th>id</th> <th>action</th> </thead> <?php foreach(\$result as \$r1){ ?> <tbody> <td><?php echo \$r1->title; ?></td> <td id="rating-<?=\$r1->id22; ?>"><?php echo \$r1- >rating; ?></td> <td name="ups" id="ups-<?=\$r1->id22; ?>"><?php echo \$r1->ups; ?></td> </pre>
--

```

    <td      name="downs"      id="downs-<?=$r1->id22;
?>"><?php echo $r1->downs; ?></td>
    <td      name="id1"      id="id1"><?php echo $r1->id22;
?></td>

```

Kode pada gambar 5.9 digunakan untuk menampilkan nama tabel yang akan ditampilkan pada halaman fitur voting. Table action merupakan tombol up dan down untuk pengguna melakukan voting. Table ratingdowns merupakan table untuk jumlah rating down yang telah divoting oleh pengguna. Table ratingups merupakan table untuk jumlah rating up yang telah divoting oleh pengguna. Table rating merupakan table untuk menampilkan perhitungan voting dari pengguna. Table title merupakan table nama aplikasi yang telah diambil dari GooglePlay. Kode pada Tabel 5. juga melakukan foreach untuk melakukan panggilan looping terhadap setiap isi table-tabel yang telah disebutkan.

Tabel 5.10 Implementasi Wilson Score Interval

```

home.blade.php
function confidence(ups, downs){
    var n = ups + downs;
    // console.log(ups);
    // console.log(downs);
    if(n == 0){
        console.log('masuk');
        return 0;
    }else{
        var z = 1.96;
        var phat = 1.0*ups / n;
        var hasil;
        // var akar = (phat*(1-phat)+(z*z/(4*n))/n);
        hasil = (((phat + z*z/(2*n))- (z * Math.sqrt((phat*(1-
phat)+(z*z/(4*n)))/n)))/(1+z*z/n));

        console.log(hasil);
        return hasil;
    }
}

```

Kode pada Tabel 5.10 merupakan rumus perhitungan dari Wilson Score Interval. Dimana function diatas dilakukan validasi dimana langkah pertama mengambil nilai up dan down dari hasil inputan kemudian dilakukan validasi. Kemudian dilakukan perhitungan dengan Wilson score interval berdasarkan jumlah up dan down yang telah diinputkan.

Tabel 5.11 Kode Untuk Menghitung Jumlah Voting Up

```

home.blade.php
function updateForm(myId) {

    var id22 = myId;
    var ups1 = document.getElementById('ups-
'+myId);
    var ups = parseFloat(ups1.innerHTML)+1.0;
    console.log(ups);
    var downs1 = document.getElementById('downs-
'+myId);
    var downs = parseFloat(downs1.innerHTML);
    console.log(downs);

    var hasil_conf = confidence(ups,downs);
    var url_do = "ups1";
    console.log( );
    $.get(url_do,{

        hasil_conf: hasil_conf,
        id22 : id22,
        ups : ups
    }, function (data) {

        $("#isitabel").empty();
        $("#isitabel").html(data);
    });
}

```

Kode pada Tabel 5.11 untuk menghitung voting up yang telah divoting oleh pengguna. Kemudian akan dipanggil oleh function confidence untuk dihitung menggunakan rumus.

Tabel 5.12 Kode Untuk Menghitung Jumlah Voting Down

```

home.blade.php
function updateForma(myId) {
    var id22 = myId;
    var ups1 = document.getElementById('ups-
'+myId);
    var ups = parseFloat(ups1.innerHTML);
    var downs1 = document.getElementById('downs-
'+myId);
    var downs = parseFloat(downs1.innerHTML)+1;

    var hasil_conf = confidence(ups,downs);
    console.log(parseFloat(hasil_conf));
    var url_do = "downs1";
    $.get(url_do,{

        hasil_conf: hasil_conf,
        id22 : id22,
        downs : downs
    }, function (data) {

        $("#isitabel").empty();
        $("#isitabel").html(data);
    });
}

```

Kode pada Tabel 5.12 untuk menghitung voting down yang telah divoting oleh pengguna. Kemudian akan dipanggil oleh function confidence untuk dihitung menggunakan rumus.

Tabel 5.13 Kode Untuk Menampilkan Hasil Voting

```

Awal.blade.php
<?php foreach($result as $r1){ ?>
    <tbody>
        <td><?php echo $r1->title; ?></td>
        <td id="rating-<?=$r1->id22; ?>"><?php echo $r1-
>rating; ?></td>
        <td name="ups" id="ups-<?=$r1->id22; ?>"><?php
echo $r1->ups; ?></td>
        <td name="downs" id="downs-<?=$r1->id22;
?>"><?php echo $r1->downs; ?></td>
        <td name="id1" id="id1"><?php echo $r1->id22;
?></td>

        <td>
            <!--      <input type="text" name="id1" id="id1"
value="<?php echo $r1->id22; ?>">-->
            <!--      <input type="text" name="ups" id="ups"
value="<?=$r1->ups; ?>">-->
            <!--      <input type="text" name="downs" id="downs"
value="<?=$r1->downs; ?>">-->
            <button onClick="updateForm(<?=$r1->id22; ?>)">
up </button>
            <button onClick="updateForma(<?=$r1->id22;?>)">
down </button>
        </td>
    </tbody>
<?php
}
?>

```

Kode pada Tabel 5.13 untuk menampilkan voting yang telah di dilakukan oleh pengguna yang akan ditampilkan atau dipanggil pada halaman `home.blade.php`.

Tabel 5.14 Kode Untuk Menampilkan Data Dari Basis Data

```
UcanController.php
class UcanController extends Controller
{
    public function index(){
        $sucan = DB::select("SELECT distinct tes.id22, tes.id,
tes.rating, applications.title, tes.ups, tes.downs
        FROM tes INNER JOIN applications ON tes.id =
applications.id
        where applications.genreId = 'music_and_audio'
        order by tes.rating desc, applications.maxInstalls desc,
applications.score desc, applications.reviews desc
        limit 100");

        return view('home',['result' => $sucan]);
    }
}
```

Kode pada Tabel 5.14 untuk menampilkan data dari basis data dengan kondisi dimana kategorinya adalah music and audio.

Tabel 5.15 Kode Untuk Melakukan Perhitungan Up

```
UcanController.php
public function ups1(){
    $hasil_conf = $_GET['hasil_conf'];
    $id22 = $_GET['id22'];
    $ups = $_GET['ups'];
    $ups = $ups;
    $master = DB::table('tes')->where('id22', $id22)-
>update(['ups' => $ups,'rating'=> $hasil_conf]);
    $sukan = DB::select("SELECT distinct tes.id22, tes.id,
tes.rating, applications.title, tes.ups, tes.downs FROM tes
INNER JOIN applications ON tes.id = applications.id where
applications.genreId = 'music_and_audio' order by tes.rating
desc, applications.maxInstalls desc, applications.score desc,
applications.reviews desc limit 100");

    return view('awal',['result' => $sukan]);
}
```

Kode pada Tabel 5.15 untuk melakukan perhitungan up yang telah diupdate kemudian ditampilkan pada view awal lalu akan dipanggil pada home.blade.php.

Tabel 5.16 Kode Untuk Melakukan Perhitungan Down

```

UcanController.php
public function downs1(){
    $hasil_conf = $_GET['hasil_conf'];
    $id22 = $_GET['id22'];
    $downs = $_GET['downs'];
    $downs = $downs;
    $master = DB::table('tes')->where('id22', $id22)-
>update(['downs' => $downs,'rating'=> $hasil_conf]);
    $sucan = DB::select("SELECT distinct tes.id22, tes.id,
tes.rating, applications.title, tes.ups, tes.downs FROM tes
INNER JOIN applications ON tes.id = applications.id where
applications.genreId = 'music_and_audio' order by tes.rating
desc, applications.maxInstalls desc, applications.score desc,
applications.reviews desc limit 100");

    return view('awal',['result' => $sucan]);
}

```

Kode pada gambar ___ untuk melakukan perhitungan down yang telah diupdate kemudian ditampilkan pada view awal lalu akan dipanggil pada home.blade.php.

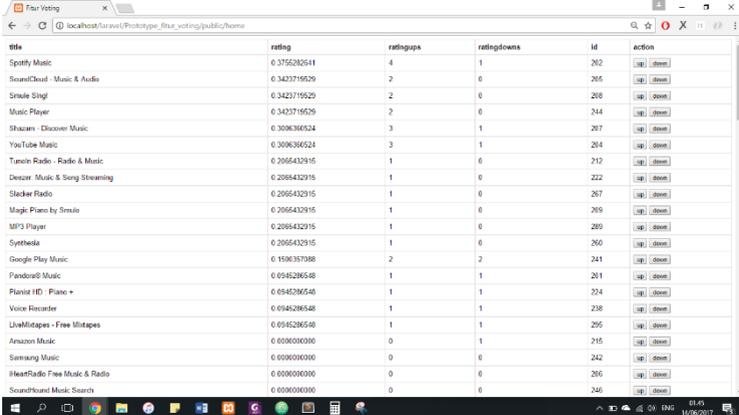
BAB VI HASIL DAN PEMBAHASAN

Pada tahapan ini berisi hasil dan pembahasan dari implementasi penelitian. Hasil yang didapatkan yaitu prototipe fitur voting menggunakan metode Wilson score interval dan perbandingan metode average rating dengan metode Wilson score interval serta Google Play Store dengan prototipe fitur voting.

6.1 Hasil Prototipe Aplikasi

Hasil prototipe aplikasi berupa antar muka pengguna dengan kapabilitas terbatas hanya pada fitur voting serta melihat hasil rating produk yang telah di voting oleh pengguna. Terdapat satu tampilan antar muka yaitu halaman fitur voting. Berikut adalah hasil dan pembahasan mengenai masing-masing fitur halaman.

6.1.1 Halaman Fitur Voting



The screenshot shows a web browser window titled 'Fitur Voting' with the URL 'localhost:7441/Prototype_Fitur_voting/public/home'. The page displays a table with the following columns: 'title', 'rating', 'ratingsups', 'ratingdowns', 'id', and 'action'. The table lists various music tracks and their corresponding voting data.

title	rating	ratingsups	ratingdowns	id	action
Spotify Music	0.375292541	4	1	262	Up Down
SoundCloud - Music & Audio	0.3423719529	2	0	265	Up Down
Smule Sing!	0.3423719529	2	0	268	Up Down
Music Player	0.3423719529	2	0	244	Up Down
Shazam - Discover Music	0.3096366524	3	1	267	Up Down
YouTube Music	0.3096366524	3	1	264	Up Down
TuneIn Radio - Radio & Music	0.2065432315	1	0	212	Up Down
Deezer Music & Song Streaming	0.2065432315	1	0	222	Up Down
Stacker Radio	0.2065432315	1	0	267	Up Down
Magic Piano by Smule	0.2065432315	1	0	269	Up Down
MP3 Player	0.2065432315	1	0	289	Up Down
Synthesia	0.2065432315	1	0	260	Up Down
Google Play Music	0.15963573988	2	2	241	Up Down
Pandora® Music	0.0546296548	1	1	261	Up Down
Planzit HD - Piano	0.0546296548	1	1	224	Up Down
Voice Recorder	0.0546296548	1	1	238	Up Down
LiveMitates - Free Mitapes	0.0546296548	1	1	295	Up Down
Amazon Music	0.0090006000	0	1	215	Up Down
Samsung Music	0.0090006000	0	0	242	Up Down
iHeartRadio Free Music & Radio	0.0090006000	0	0	266	Up Down
SoundCloud Music Search	0.0090006000	0	0	245	Up Down

Gambar 6.1 Halaman Fitur Voting

Halaman pada Gambar 6.1 merupakan halaman fitur voting dimana daftar aplikasi yang diperoleh dari hasil *Web Scrapping* pada *Google Play Store* dengan kategori *Music and Audio*. Pada halaman ini pengguna dapat melakukan *voting up* dan *down*

pada aplikasi yang diinginkan. Dari hasil *voting* yang dilakukan pengguna dapat melihat jumlah *voting* yang ada untuk setiap aplikasi. Berdasarkan hasil *voting* tersebut nantinya akan ditampilkan *rating* yang berasal dari hasil perhitungan jumlah *voting up* dan *down* menggunakan perhitungan *Wilson Score Interval*. Dari hasil perhitungan yang dilakukan akan didapatkan urutan atau ranking aplikasi berdasarkan perhitungan *rating*.

6.2 Hasil Analisis Perbandingan

Pada tahapan ini akan dilakukan perbandingan antara dua metode perhitungan yaitu average rating dengan *Wilson score interval* dan perbandingan hasil pencarian pada kategori *music and audio Google Play Store* dengan Prototipe Fitur Voting.

6.2.1 Average Rating Dengan Wilson Score Interval (Real Time)

Pada eksperimen kali ini dilakukan perbandingan dari dua metode dengan melakukan survey kepada 10 pengguna android. Survey dilakukan dengan menggunakan prototipe fitur voting yang dirancang dimana pengguna melakukan voting pada aplikasi yang diinginkan terlebih dahulu dengan ketentuan maksimal 3 voting up atau down untuk satu aplikasi. Voting yang dilakukan tidak membatasi jumlah aplikasi untuk diberikan vote. Berdasarkan eksperimen yang dilakukan terhadap 10 aplikasi yang sudah ditentukan, berikut ini merupakan hasil dari perhitungan dari kedua metode.

Tabel 6. 1 Hasil Perhitungan Average Rating dan Wilson Score Interval

Nama Aplikasi	Jumlah Voting		Hasil Perhitungan Wilson	Rank	Hasil Perhitungan Avera	Rank
	Up	Down				

			Score Interval		ge Rating	
Spotify Music	8	2	0,49	1	0,80	4
SoundCloud - Music & Audio	5	1	0,43	2	0,83	3
Smule Sing!	6	2	0,40	3	0,75	5
Music Player	7	3	0,39	4	0,70	7
Shazam - Discover Music	2	0	0,34	5	1	1
YouTub e Music	3	1	0,30	6	0,75	6
TuneIn Radio - Radio & Music	1	0	0,20	7	1	2
Deezer: Music & Song Streamin g	2	2	0,15	8	0,50	8
Google Play Music	1	1	0,09	9	0,50	9
Pandora ® Music	0	1	0	10	0	10

Berdasarkan eksperimen diatas dilakukan perhitungan rating dengan jumlah voting yang sama untuk mengetahui ranking atau urutan aplikasi berdasarkan kedua metode. Dari table diatas voting dilakukan diwaktu yang bersamaan dan diketahui bahwa urutan atau ranking aplikasi yang ditampilkan berbeda. Dari table diatas diambil tiga contoh aplikasi (spotify, tune in radio, dan shazam) diketahui jumlah voting yang dimiliki oleh aplikasi spotify adalah 10 vote dengan rincian 8 vote up dan 2 vote down, jumlah voting aplikasi tune in adalah 1 vote dengan rincian 1 vote up dan 0 down, dan jumlah voting aplikasi shazam adalah 2 vote dengan rincian 2 vote up dan 0 down. Berdasarkan data diatas, kita dapat menilai bahwa pengguna menilai aplikasi spotify lebih baik dibandingkan aplikasi lainnya dengan melihat jumlah voting up yang dimiliki aplikasi spotify lebih banyak dibandingkan dua aplikasi lainnya (8 vote up). Namun berdasarkan perhitungan dengan menggunakan metode average rating, aplikasi spotify dianggap tidak lebih baik dibandingkan aplikasi Shazam dan Tune Inn. Perhitungan dengan average rating menempatkan aplikasi shazam pada urutan pertama, dan tune in pada urutan kedua. Average Rating hanya memperhitungkan hasil tetapi tidak memperhitungkan jumlah voting pengguna. Sedangkan berdasarkan metode Wilson score interval, aplikasi spotify berada pada urutan pertama dan sesuai dengan pilihan pengguna pada saat melakukan voting. Berdasarkan preference pengguna perhitungan Average Rating tidak lebih bagus daripada Wilson Score Interval dikarenakan jumlah voting pengguna pada suatu aplikasi kurang diperhitungkan sehingga urutan yang ditampilkan kurang cocok dengan hasil voting yang sudah dilakukan.

Berikut ini dilakukan perhitungan secara manual Average Rating dengan Wilson Score Interval, sampel yang diambil urutan pertama pada perhitungan Average Rating yaitu aplikasi

Shazam. Berdasarkan teori yang telah dijelaskan pada bab sebelumnya perhitungan menggunakan average rating dilakukan dengan membagi jumlah voting up dengan total voting yang dimiliki setiap aplikasi tanpa memperhitungkan *Confidence Level* atau selang kepercayaan dari jumlah voting yang dimiliki. Sedangkan metode *Wilson Score Interval* mempertimbangkan nilai *confidence level* sebesar 95% dari jumlah voting

Berikut ini merupakan hasil perhitungan yang diimplementasikan secara manual dengan menggunakan kedua metode:

- Nama Aplikasi : Shazam – Discover Music
- Jumlah Voting Up : 2
- Jumlah Voting Down : 0

Average Rating

$$\text{Score} = \frac{\text{Positive Ratings}}{\text{Total Ratings}}$$

$$\text{Score} = \frac{2}{2} = 1$$

Dari perhitungan *Average Rating* didapatkan hasil sebesar 1.

Wilson Score Interval

$$\text{score} = \frac{\hat{p} + \frac{z_{1-\alpha/2}^2}{2n} - z_{1-\alpha/2}^2 \sqrt{\hat{p}(-\hat{p}) + \frac{z_{1-\alpha/2}^2}{4n}}/2}{1 + z_{1-\alpha/2}^2/n}$$

Diketahui:

$$n = \text{Total Ratings} = 2$$

$$\hat{p} = 1 \times \text{Positive Ratings/Total Ratings} = 1 \times 2/2 = 1$$

$$z_{1-\alpha/2} = 1,96 \text{ (Confidence Level 95\%)}$$

$$\text{score} = \frac{1 + \frac{1,96^2}{2 \times 2} - 1,96 \sqrt{1(1-1) + \frac{1,96^2}{4 \times 2}}}{1 + 1,96^2/2} = 0,3423719529$$

Dari perhitungan *Wilson Score Interval* didapatkan hasil sebesar 0,34.

Berdasarkan hasil perhitungan diatas diketahui bahwa kedua peritungan memiliki jumlah voting yang sama namun hasil yang didapatkan dari kedua perhitungan berbeda. Hal ini disebabkan karena metode *Wilson score interval* mempertimbangkan nilai *Confidence Level* sebesar 95% sehingga hasil perhitungan metode ini lebih baik dibandingkan perhitungan *average rating*.

Perspektif Pengguna

Setelah melakukan eksperimen diatas, pengguna kemudian diminta untuk melakukan penilaian terhadap hasil dari eksperimen yang dilakukan. Dari hasil eksperimen yang dilakukan diperoleh hasil ranking urutan aplikasi dari kedua metode

Tabel 6. 2 Ranking Aplikasi Berdasarkan Kedua Metode

No.	Ranking Aplikasi Wilson Score Interval	Ranking Aplikasi Average Rating
1.	Spotify Music	Shazam - Discover Music
2.	SoundCloud - Music & Audio	TuneIn Radio - Radio & Music
3.	Smule Sing!	SoundCloud - Music & Audio

4.	Music Player	Spotify Music
5.	Shazam - Discover Music	Smule Sing!
6.	YouTube Music	YouTube Music
7.	TuneIn Radio - Radio & Music	Music Player
8.	Deezer: Music & Song Streaming	Deezer: Music & Song Streaming
9.	Google Play Music	Google Play Music
10.	Pandora® Music	Pandora® Music

Dari table diatas pengguna diminta untuk melakukan penilaian untuk menentukan:

1. Urutan rangking yang paling sesuai (Wilson Score Interval atau Average Rating)
2. Seberapa tepat urutan rangking menurut pendapat pengguna

Untuk mengetahui hasil pernyataan diatas, dilakukan survey dengan menggunakan kuesioner yang diukur dengan skala likert.

Peer Review 02

Gambar dibawah ini merupakan urutan hasil voting yang telah dilakukan sebelumnya berdasarkan perhitungan dari kedua metode

Rangking Aplikasi

No.	Rangking Aplikasi Wilson Score Interval	Rangking Aplikasi Average Rating
1.	Spotify Music	Shazam - Discover Music
2.	SoundCloud - Music & Audio	TuneIn Radio - Radio & Music
3.	Smule Sing!	SoundCloud - Music & Audio
4.	Music Player	Spotify Music
5.	Shazam - Discover Music	Smule Sing!
6.	YouTube Music	YouTube Music
7.	TuneIn Radio - Radio & Music	Music Player
8.	Deezer: Music & Song Streaming	Deezer: Music & Song Streaming
9.	Google Play Music	Google Play Music
10.	Pandora® Music	Pandora® Music

Dari kedua urutan rangking tersebut, manakah urutan rangking yang paling sesuai menurut anda *

Wilson Score Interval

Average Rating

Seberapa tepatkah hasil urutan rangking menurut pendapat anda *

1 2 3 4

Sangat Tidak Sangat Tepat

Tepat

Gambar 6.2 Kuesioner Perspektif Pengguna 01

Variabel pada penelitian ini dinilai dengan melihat dari nilai rata-rata dan memberi arti nilai tersebut dengan membuat kriteria berdasarkan pada interval kelas rata-rata. Interval kelas yang digunakan berdasarkan pada rumus:

$$Interval = \frac{Nilai Tertinggi - Nilai Terendah}{Banyaknya Kelas}$$

$$Interval = \frac{4 - 1}{4} = 0,75$$

Dari perhitungan diatas sehingga didapatkan rentang skala penilaian setiap variabel berikut:

Tabel 6. 3 Skala Penilaian

Interval Rata-rata	Penilaian
$1,00 \leq x \leq 1,75$	Sangat Tidak Tepat
$1,75 \leq x \leq 2,50$	Tidak Tepat
$2,50 \leq x \leq 3,25$	Tepat
$3,25 \leq x \leq 4,00$	Sangat Tepat

Dari hasil review yang telah dilakukan oleh 10 pengguna didapat hasil bahwa seluruh pengguna memilih Wilson Score Interval menampilkan urutan ranking yang paling sesuai. Kemudian untuk menentukan seberapa tepat urutan ranking menurut pendapat pengguna dilakukan penilaian menggunakan skala likert (1-4) dan didapatkan hasil bahwa 8 pengguna menilai hasil yang ditampilkan tepat dan 2 pengguna lainnya menilai sangat tepat.

Tabel 6. 4 Hasil Kuesioner 01

Skala Penilaian	Sangat Tidak Tepat	Tidak Tepat	Tepat	Sangat Tepat	Rata-Rata
Jumlah	0	0	8	2	3,20

Dari hasil diatas dengan nilai rata-rata sebesar 3,20 yang berada pada skala penilaian tepat, oleh karena itu menurut penilaian perspektif pengguna metode Wilson Score Interval menampilkan urutan aplikasi yang tepat menurut pengguna.

6.2.2 Eksperimen Desain Berdasarkan User Review dan User Experience

Pada eksperimen selanjutnya dilakukan voting pada 10 aplikasi acak untuk kategori music pada 10 pengguna google playstore. Pada eksperimen ini dilakukan dua jenis voting, yaitu voting berdasarkan review (deskripsi aplikasi, review pengguna, rating aplikasi pada google playstore) dan voting berdasarkan user experience (pengguna mencoba 10 aplikasi yang ditentukan). Dua voting ini tidak dilakukan secara bersamaan dan pada metode Wilson Score Interval saat melakukan perhitungan fase kedua tidak memperhitungkan hasil vote fase pertama dimana voting berdasarkan user experience (fase kedua) dilakukan 3 hari setelah voting berdasarkan review (fase pertama). Pada hasil akhir eksperimen ini, hasil perhitungan rating berdasarkan metode average rating melibatkan keseluruhan hasil voting (2 fase), sedangkan metode Wilson score interval hanya menggunakan jumlah voting berdasarkan user experience (hanya fase kedua). Hal ini dikarenakan metode Wilson score interval tidak menghitung rating pada fase sebelumnya, dan hanya menggunakan jumlah voting pada fase saat ini.

Tabel dibawah ini menampilkan hasil voting pengguna yang melakukan voting berdasarkan review dari aplikasi yang ada pada google playstore.

Tabel 6. 5 Hasil Voting Kedua Metode Berdasarkan Review Aplikasi (Fase Pertama Kedua Metode)

Nama Aplikasi	Jumlah Voting		Rating berdasarkan wilson	Ranking aplikasi berdasarkan wilson	Rating berdasarkan average	Ranking aplikasi berdasarkan average
	up	down				
Soundcloud	8	0	0.67556	1	1	1
Soundhound	5	0	0.56550	2	1	2
Shazam	3	0	0.43849	3	1	3
TuneInn Radio	3	0	0.43849	4	1	4
Smule Sing	4	1	0.37552	5	0.8	8
Music Player	2	0	0.34237	6	1	5
Deezer	2	1	0.20765	7	0.67	9
My Mixtapez Music	1	0	0.20654	8	1	6
Synthesia	1	0	0.20654	9	1	7
Ares Musica	0	0	0.00000	10	0	10

Berikut ini merupakan hasil perhitungan rating dengan menggunakan kedua metode. Pada fase pertama diketahui kedua metode menyebutkan aplikasi Soundcloud berada pada urutan pertama, Soundhound pada urutan kedua, dan ares musica berada pada urutan terakhir. Berdasarkan metode Wilson aplikasi synthesia menduduki urutan ke 9 dimana berdasarkan metode average aplikasi ini berada pada urutan ke 7.

Dalam selang waktu 3 hari, selanjutnya dilakukan voting untuk fase kedua dimana user melakukan voting setelah menggunakan 10 aplikasi yang sudah ditentukan. Berikut ini merupakan hasil voting untuk fase kedua menggunakan average rating dimana rating fase kedua memperhitungkan jumlah voting pada fase pertama.

Tabel 6.6 Hasil Perhitungan Average Rating Pada Fase Pertama dan Kedua (User Experience)

Nama Aplikasi	Jumlah Voting		Rating berdasarkan fase pertama	Ranking aplikasi	Jumlah voting		Rating berdasarkan fase kedua	Ranking aplikasi
	up	down			up	down		
Soundcloud	8	0	1	1	14	2	0.875	5
Soundhound	5	0	1	2	13	0	1	1
Shazam	3	0	1	3	4	3	0.571	8
TuneInn Radio	3	0	1	4	5	0	1	2
Smule Sing	4	1	0.8	8	6	4	0.60	7
Music Player	2	0	1	5	3	0	1	3
Deezer	2	1	0.67	9	10	1	0.9	4
My Mixtapez Music	1	0	1	6	3	3	0.5	9
Synthesia	1	0	1	7	4	2	0.67	6
Ares Musica	0	0	0	10	0	5	0	10

Berdasarkan table diatas diketahui rangking aplikasi di 2 fase yang berbeda. Pada fase pertama diketahui bahwa aplikasi Soundhound menempati urutan kedua sedangkan pada fase kedua aplikasi ini menempati urutan pertama. Dapat dilihat dari hasil diatas tidak terjadi kenaikan rangking yang signifikan pada aplikasi berdasarkan metode average rating. Rangking yang dimiliki setiap aplikasi tidak jauh berbeda dari fase sebelumnya dikarenakan voting yang dimiliki aplikasi di fase pertama juga digunakan pada saat menghitung rating di fase kedua. Sehingga aplikasi yang memiliki jumlah up votes yang lebih banyak dibandingkan dengan aplikasi lainnya akan berada pada posisi paling atas. Hal ini menyebabkan dengan menggunakan average rating aplikasi yang populer akan terus populer seiring berjalannya waktu.

Berikut ini merupakan hasil voting untuk fase kedua menggunakan Wilson Score Interval dimana rating fase kedua tidak memperhitungkan jumlah voting pada fase pertama.

Tabel 6.7 Hasil Perhitungan Wilson Score Interval Pada Fase Pertama dan Kedua (User Experience)

Nama Aplikasi	Jumlah Voting		Rating berdasarkan fase pertama	Ranking aplikasi	Jumlah voting		Rating berdasarkan fase kedua	Ranking aplikasi
	up	down			up	down		
Soundcloud	8	0	0.67556	1	6	2	0.409269	3
Soundhound	5	0	0.56550	2	7	0	0.645661	2
Shazam	3	0	0.43849	3	1	3	0.045586	9
TuneInn Radio	3	0	0.43849	4	2	0	0.342371	4
Smule Sing	4	1	0.37552	5	2	3	0.117618	7
Music Player	2	0	0.34237	6	1	0	0.206543	6
Deezer	2	1	0.20765	7	8	0	0.675584	1
My Mixtapez Music	1	0	0.20654	8	2	3	0.117618	8
Synthesia	1	0	0.20654	9	3	1	0.300636	5
Ares Musica	0	0	0.00000	10	0	5	0.000000	10

Tabel diatas merupakan hasil perbandingan rating dan ranking aplikasi pada fase pertama dan kedua. Pada fase pertama diketahui bahwa aplikasi Soundcloud menempati urutan pertama namun pada fase kedua aplikasi ini menempati urutan ke 3. Aplikasi Deezer yang berada pada urutan ke 7 di fase pertama mengalami kenaikan yang signifikan di fase kedua yang membuat aplikasi ini menempati urutan pertama. Dapat dilihat juga aplikasi Shazam yang berada pada urutan ke 3 di fase pertama namun untuk fase kedua aplikasi ini mengalami penurunan yang signifikan sehingga menempati urutan ke 9. Dari hasil diatas diketahui bahwa jumlah voting yang dimiliki oleh aplikasi pada fase pertama tidak diperhitungkan pada fase kedua.

Berdasarkan hasil analisis diatas pada metode Wilson Score Interval di fase kedua membuat perubahan urutan secara signifikan, aplikasi yang benar-benar tren pada masanya berada pada posisi teratas dapat turun secara signifikan pada fase selanjutnya, begitu pula dengan aplikasi yang biasa saja dapat mengalami kenaikan yang signifikan menjadi urutan yang pertama.

Tabel 6. 8 *Rangking Aplikasi dari kedua metode*

Nama Aplikasi	Ranking Wilson fase pertama	Ranking Wilson fase kedua	Rangking Average fase pertama	Rangking Average fase kedua
Soundcloud	1	3	1	5
Soundhoun d	2	2	2	1
Shazam	3	9	3	8
TuneInn Radio	4	4	4	2
Smule Sing	5	7	8	7
Music Player	6	6	5	3

Deezer	7	1	9	4
My Mixtapez Music	8	8	6	9
Synthesia	9	5	7	6
Ares Musica	10	10	10	10

Perspektif Pengguna

Setelah melakukan eksperimen diatas, pengguna diminta untuk melakukan penilaian terhadap eksperimen yang telah dilakukan. Pengguna diberikan riwayat voting yang dilakukan dikedua fase dari riwayat tersebut pengguna akan menilai factor yang mempengaruhi pilihannya. Faktor waktu yang dimaksud adalah selang waktu dari fase pertama ke fase kedua, sedangkan faktor knowledge adalah pengetahuan pengguna terhadap suatu aplikasi Faktor tersebut yaitu:

1. Faktor waktu mempengaruhi perubahan pilihan pengguna pada fase pertama dan kedua
2. Faktor *knowledge* mempengaruhi perubahan pilihan pengguna pada fase pertama dan kedua

Untuk mengetahui hasil pernyataan diatas, dilakukan survey dengan menggunakan kuesioner yang diukur dengan skala likert.

Peer Review 01

Faktor waktu mempengaruhi perubahan pilihan yang anda lakukan pada fase pertama ke fase kedua *

1 2 3 4

Sangat Tidak Setuju Sangat Setuju

*Jika tidak setuju berikan alasannya

Your answer

Faktor knowledge mempengaruhi perubahan pilihan yang anda lakukan pada fase pertama ke fase kedua *

1 2 3 4

Sangat Tidak Setuju Sangat Setuju

*Jika tidak setuju berikan alasannya

Your answer

Gambar 6.3 Kuesioner Perspektif Pengguna 02

Dari hasil penilaian yang dilakukan 10 pengguna untuk faktor waktu dengan menggunakan skala likert (1-4), 2 pengguna menilai tidak setuju, 7 menilai setuju dan 1 menilai sangat setuju bahwa factor waktu mempengaruhi perubahan pilihan pengguna. Berikut ini merupakan hasil kuesioner berdasarkan factor waktu mempengaruhi perubahan pilihan pengguna.

Tabel 6. 9 Hasil Kuesioner Faktor Waktu

Skala Penilaian	Sangat Tidak Tepat	Tidak Tepat	Tepat	Sangat Tepat	Rata-Rata
-----------------	--------------------	-------------	-------	--------------	-----------

Jumlah	0	2	7	1	2,90
--------	---	---	---	---	------

Dari hasil kuesioner diatas didapatkan hasil sebesar 2,90, yang berarti berada pada skala penilaian tepat. Alasan pengguna yang menilai tidak setuju bahwa factor waktu tidak berpengaruh dikarenakan pengguna menilai factor selera dan informasi fitur dan factor user experience yang mempengaruhi perubahan pilihan pengguna.

Selanjutnya untuk factor knowledge dengan menggunakan skala likert (1-4) diketahui 8 pengguna memilih setuju dan 2 pengguna memilih sangat setuju.

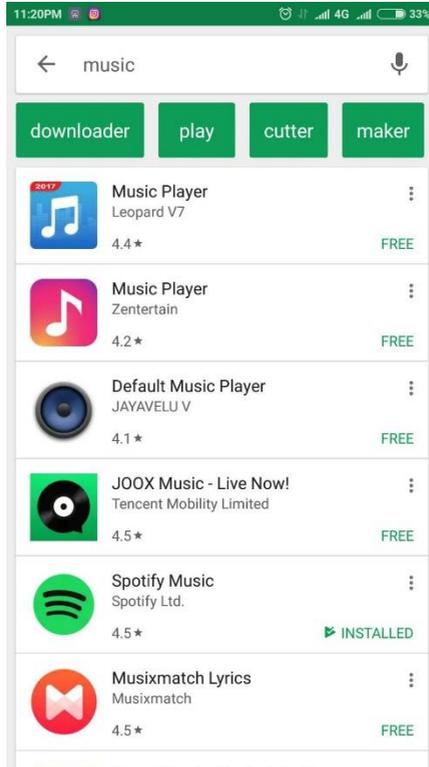
Tabel 6. 10 Hasil Kuesioner Faktor Knowledge

Skala Penilaian	Sangat Tidak Tepat	Tidak Tepat	Tepat	Sangat Tepat	Rata-Rata
Jumlah	0	0	8	2	3,20

Dari hasil kuesioner diatas didapatkan hasil sebesar 3,20 yang berarti berada pada skala penilaian tepat. Dapat disimpulkan penilaian berdasarkan perspektif pengguna ditarik kesimpulan bahwa 72,5% pengguna menyebutkan bahwa factor waktu berpengaruh terhadap perubahan pilihan pengguna, dan untuk factor knowledge 80% pengguna setuju bahwa factor ini mempengaruhi perubahan pengguna.

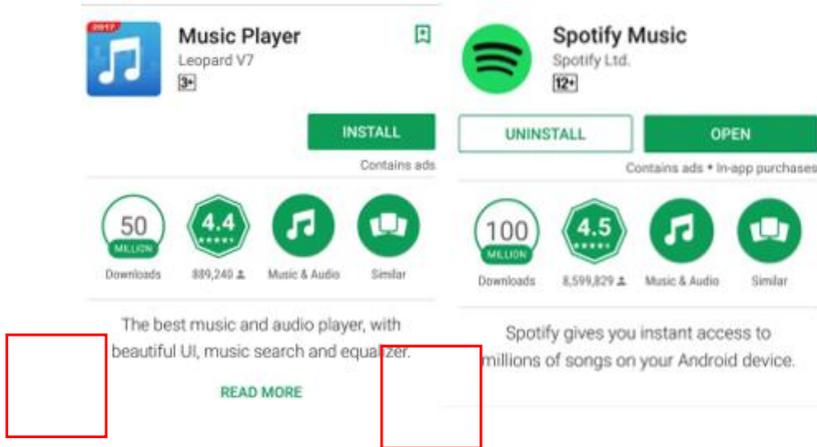
6.2.3 Google Play Store Dengan Prototipe Fitur Voting

Pada tahapan ini dilakukan uji coba perbandingan hasil pencarian Google Play Store dengan Prototipe Fitur Voting pada kategori Music. Uji coba dilakukan pada 20 pengguna dengan aplikasi yang ditampilkan yaitu 5 aplikasi berdasarkan pencarian pada google playstore



Gambar 6.4 Tampilan hasil pencarian aplikasi kategori musik pada Googleplay

Pada gambar 6.2 merupakan hasil pencarian kategori music dari GooglePlay via Mobile. Berdasarkan hasil diatas ditampilkan aplikasi music yang dimana hasil tersebut diurutkan secara acak. Seringkali pengguna akan memilih untuk mendownload aplikasi yang berada pada urutan pertama atau mempertimbangkan rating yang dimiliki aplikasi tersebut.



Gambar 6.5 Perbandingan Aplikasi Music Player dan Spotify

Dapat dilihat pada gambar diatas hasil pencarian pada google play menunjukkan bahwa urutan music player berada lebih atas daripada spotify. Berdasarkan hasil voting diketahui bahwa pengguna lebih memilih aplikasi spotify yang berada pada urutan kelima dibandingkan music player yang berada pada urutan pertama berdasarkan hasil pencarian pada google playstore. Hal ini dapat dibuktikan dari jumlah unduh, rating, dan jumlah pengguna yang melakukan rating pada aplikasi tersebut. Seharusnya google play menampilkan aplikasi spotify diatas music player karena aplikasi ini yang lebih banyak digunakan oleh pengguna.

title	rating	ratingups	ratingdowns	id	action
Spotify Music	0.7008472464	9	0	202	<input type="button" value="up"/> <input type="button" value="down"/>
SoundCloud - Music & Audio	0.4384939196	3	0	205	<input type="button" value="up"/> <input type="button" value="down"/>
Smule Sing!	0.4384939196	3	0	208	<input type="button" value="up"/> <input type="button" value="down"/>
Shazam - Discover Music	0.3423719529	2	0	207	<input type="button" value="up"/> <input type="button" value="down"/>
Music Player	0.3423719529	2	0	244	<input type="button" value="up"/> <input type="button" value="down"/>
Google Play Music	0.0000000000	0	0	241	<input type="button" value="up"/> <input type="button" value="down"/>

Gambar 6.6 Tampilan aplikasi musik pada Prototipe Fitur Voting

Pada gambar 6.4 merupakan hasil voting yang dilakukan pada prototipe fitur voting. Berdasarkan gambar 6.3 diketahui bahwa urutan aplikasi yang ditampilkan ketika melakukan pencarian kategori music pada playstore, urutan yang ditampilkan berbeda dengan hasil voting yang dilakukan kepada pengguna. Dengan memanfaatkan prototipe yang dirancang, pengguna diberikan kebebasan untuk memilih aplikasi yang biasa mereka gunakan. Dari hasil voting yang dilakukan, diketahui bahwa aplikasi spotiy merupakan aplikasi yang sering digunakan untuk kategori music. Berdasarkan hasil voting ini dapat diketahui bahwa aplikasi spotify lebih sering digunakan pengguna, namun berdasarkan hasil pencarian yang dilakukan pada playstore (Gambar 6.2), aplikasi spotify ini tidak menduduki urutan pertama. Seharusnya aplikasi yang lebih sering digunakan oleh pengguna berada pada urutan pertama karena aplikasi ini dianggap paling relevan dengan kategori ini sebab aplikasi ini lebih sering digunakan oleh pengguna. Berdasarkan penjelasan diatas, pengguna menilai bahwa prototipe ini lebih baik karena urutan aplikasi yang ditampilkan sesuai dengan apa yang mereka nilai, pengguna juga menilai bahwa prototipe yang menggunakan metode Wilson score interval lebih sesuai dalam menampilkan aplikasi yang paling relevan dibandingkan

dengan hasil pencarian kategori music yang ditampilkan pada google play.

Perspektif Pengguna

Setelah melakukan perbandingan pada tampilan Prototipe Fitur Voting dan Google Play, dilakukan penilaian perspektif pengguna terhadap eksperimen yang dilakukan. Dari hasil eksperimen yang dilakukan diperoleh hasil urutan aplikasi dari Prototipe Fitur Voting dan Google Play.

Tabel 6.11 Rangkings Aplikasi pada GooglePlay dan Prototipe Fitur Voting

No.	Google Play	Prototipe Fitur Voting
1.	Music Player	Spotify Music
2.	Default Music Player	Soundcloud
3.	JOOX Music	Smule Sing!
4.	Spotify Music	Music Player
5.	Musicmatch Lyrics	Google Play Music

Dari table diatas pengguna diminta melakukan penilaian terhadap 2 faktor yaitu:

1. Dari kedua tampilan tersebut manakah urutan yang paling sesuai menurut pengguna
2. Seberapa tepat hasil urutan ranking menurut pendapat pengguna

Untuk mengetahui hasil pernyataan diatas, dilakukan survey dengan menggunakan kuesioner yang diukur dengan skala likert.

Peer Review 03

Gambar dibawah ini merupakan urutan aplikasi hasil pencarian pada kategori music.

Hasil Pencarian

No.	Google Play	Prototipe Fitur Voting
1.	Music Player	Spotify Music
2.	Default Music Player	Soundcloud
3.	JOOX Music	Smule Sing!
4.	Spotify Music	Music Player
5.	Musicmatch Lyrics	Google Play Music

Dari kedua urutan rangking tersebut, manakah urutan rangking yang paling sesuai menurut anda *

Protipe Fitur Voting

Google Play

Seberapa tepatkah hasil urutan rangking menurut pendapat anda *

1 2 3 4

Sangat Tidak Tepat Sangat Tepat

Gambar 6.7 Kuesioner Perspektif Pengguna 03

Dari penilaian yang telah dilakukan kepada 10 pengguna, seluruh pengguna memilih prototipe fitur voting menampilkan urutan yang paling sesuai menurut pengguna. Kemudian setelah pengguna menilai tampilan yang paling sesuai, pengguna diminta untuk menilai seberapa tepat hasil urutan yang ditampilkan menggunakan skala likert (1-4), 8 pengguna memilih setuju dan 2 pengguna memilih sangat setuju dengan hasil yang ditampilkan.

Tabel 6. 12 Hasil Kuesioner 03

Skala Penilaian	Sangat Tidak Tepat	Tidak Tepat	Tepat	Sangat Tepat	Rata-Rata
Jumlah	0	0	8	2	3,20

Dari hasil diatas dengan nilai rata-rata sebesar 3,20 yang berada pada skala penilaian tepat. Penilaian yang telah dilakukan dapat disimpulkan menurut perspektif pengguna protitpe fitur voting menampilkan urutan yang tepat. Dari wawancara yang dilakukan, hampir semua pengguna menggunakan aplikasi Spotify untuk mendengarkan lagu. Menurut pengguna aplikasi ini paling nyaman untuk digunakan dibandingkan dengan aplikasi music lainnya. Berdasarkan urutan hasil pencarian yang ada pada Google Play dan Prototipe Fitur Voting, aplikasi Spotify berada pada urutan pertama pada Prototipe Fitur Voting. Oleh karena itu pengguna lebih memilih Prototipe Fitur Voting karena menampilkan aplikasi yang mereka sukai pada urutan pertama.

Pada aplikasi Google Play saat ini urutan atau hasil pencarian aplikasi yang ditampilkan tidak menempatkan aplikasi yang sering digunakan oleh pengguna pada posisi teratas. Seharusnya aplikasi yang sering digunakan oleh pengguna ditampilkan terlebih dahulu dibandingkan aplikasi lainnya. Pada kondisi saat ini Google Play tidak menempatkan aplikasi spotify pada urutan pertama walaupun aplikasi ini memiliki jumlah pengguna yang lebih banyak dibandingkan aplikasi music player (dilihat dari jumlah pengguna yang mengunduh aplikasi, jumlah voting, dan jumlah rating). Seharusnya aplikasi yang sering digunakan pengguna berada pada posisi paling atas sehingga memudahkan pengguna awam dalam memperoleh

aplikasi yang paling sesuai dengan kategori tersebut. Pada prototipe fitur voting ini kemudian diterapkan implementasi dari googleplay dimana jumlah voting aplikasi spotify lebih banyak dibandingkan music player. Berdasarkan hasil tersebut didapatkan bahwa pada prototipe fitur voting ini menempatkan aplikasi spotify terlebih dahulu dibandingkan dengan music player. Hasil yang ditampilkan lebih sesuai atau relevan sebab aplikasi yang memiliki jumlah voting paling banyak dan rating paling tinggi berada pada posisi teratas, sehingga ketika pengguna melakukan pencarian aplikasi hasil pencarian aplikasi yang ditampilkan dapat memenuhi kebutuhan pengguna.

BAB VII

KESIMPULAN DAN SARAN

Pada bab ini dijelaskan mengenai kesimpulan dari hasil penelitian dan saran untuk pengembangan penelitian di kemudian hari.

7.1 Kesimpulan

Berdasarkan dengan pengerjaan tugas akhir dengan judul “Prototipe Fitur Voting Pada Hasil Pencarian Produk Perangkat Lunak Untuk Meningkatkan Relevansi Hasil Pencarian”,

- Dikembangkan prototipe yang digunakan untuk menghitung jumlah voting pengguna pada suatu aplikasi sehingga dihasilkan urutan aplikasi berdasarkan hasil perhitungan voting up dan down yang di inputkan oleh pengguna menggunakan metode perhitungan Wilson Score Interval.
- Didapatkan hasil bahwa dengan adanya sensitif waktu menggunakan metode Wilson Score Interval terjadi perubahan urutan secara signifikan, aplikasi yang benar-benar tren pada masanya berada pada posisi teratas dapat turun secara signifikan pada fase selanjutnya, begitu pula dengan aplikasi yang biasa saja dapat mengalami kenaikan yang signifikan menjadi berada pada posisi atas.
- Dari penelitian yang dilakukan didapatkan hasil bahwa metode Wilson score interval lebih relevan dikarenakan jumlah voting pengguna pada suatu aplikasi diperhitungkan sehingga urutan yang ditampilkan cocok dengan hasil voting yang sudah dilakukan dan urutan aplikasi sesuai dengan apa yang pengguna nilai sehingga aplikasi yang paling disukai atau paling

populer akan berada pada posisi teratas, selain itu urutan yang dihasilkan berdasarkan hasil perhitungan rating tidak dalam urutan yang acak sehingga hasil yang ditampilkan dapat memudahkan pengguna dalam menentukan aplikasi yang sesuai.

- Relevansi yang didapatkan dari penelitian ini dapat dilihat dari hasil perbandingan Google Play pada kategori pencarian musik dengan Prototipe Fitur Voting, pada aplikasi Google Play saat ini urutan atau hasil pencarian aplikasi yang ditampilkan tidak menempatkan aplikasi yang sering digunakan oleh pengguna pada posisi teratas. Hal ini dinilai bahwa urutan aplikasi pada Google Play dilakukan secara acak atau kurang relevan, hal ini dikarenakan Google Play tidak memperhitungkan urutan berdasarkan hasil rating, jumlah voting pengguna, dan jumlah download. Berdasarkan survey yang telah dilakukan oleh 20 pengguna, pengguna menilai bahwa Prototipe Fitur Voting yang menggunakan metode Wilson Score Interval lebih relevan karena urutan yang ditampilkan sesuai dengan perhitungan rating dan jumlah voting pengguna.

7.2 Saran

Berdasarkan penelitian pada tugas akhir ini dapat diberikan saran ataupun masukan untuk pengembangan selanjutnya, antara lain:

1. Untuk penelitian selanjutnya dapat mencari metode untuk perhitungan voting yang mempertimbangkan waktu untuk meningkatkan relevansi hasil pencarian
2. Untuk penelitian selanjutnya dapat ditambahkan fitur pop up untuk setiap aplikasi yang sudah di voting, dan ditambahkan fitur untuk menandai aplikasi yang sudah di voting.
3. Prototipe ini dapat dilanjutkan untuk langsung diimplementasikan pada website e-commerce yang memiliki daftar produk yang membutuhkan rating.
4. User Interface pada prototipe ini dapat dikembangkan lebih lanjut untuk meingkatkan kepuasan pengguna dalam menggunakan prototipe ini.

(Halaman ini sengaja dikosongkan)

DAFTAR PUSTAKA

- [1] “Indonesia Internet Users,” *Internet Live Stats*, 2017. [Online]. Available: <http://www.internetlivestats.com/internet-users/indonesia/>.
- [2] P. Roma and D. Ragaglia, “Revenue models, in-app purchase, and the app performance: Evidence from Apple’s App Store and Google Play,” *Electron. Commer. Res. Appl.*, vol. 17, pp. 173–190, 2016.
- [3] S. Amir, “How Reddit ranking algorithms work,” *Medium*, 2015. .
- [4] P. R. Suri, “An Integrated Ranking Algorithm for Efficient Information Computing in Social Networks,” *Int. J. Web Serv. Comput.*, vol. 3, no. 1, pp. 31–44, 2012.
- [5] R. Jin and L. Si, “A study of methods for normalizing user ratings in collaborative filtering,” *Proc. 27th Annu. Int. Conf. Res. Dev. Inf. Retr. - SIGIR '04*, p. 568, 2004.
- [6] S. Jaiswal and A. Kumar, “Research on Android app Vs Apple app Market: Who is leading?,” *Int. J. Eng. Comput. Sci.*, vol. 3, no. 4, pp. 5553–5556, 2014.
- [7] K. C. G. T. Laudon, *E-Commerce 2014: Business, Technology, Society 10th Edition*, 10th ed. Harlow,

England Pearson, 2014, 2013.

- [8] I. Lee, *Trends in E-Business, E-Services, and E-Commerce*. IGI GLOBAL, 2013.
- [9] T. Matthew, *php|architect's Guide to Web Scraping*. Toronto: Marco Tabini, 2010.
- [10] J. A. A. Leon;Suryayusra, "PENERAPAN TEKNIK WEB SCRAPING PADA MESIN PENCARI ARTIKEL ILMIAH," vol. 5, p. 5, 2014.
- [11] A. I. Ovi Dyantina, Mira Afrina, "Penerapan Customer Relationship Management (CRM) Berbasis Web (Studi Kasus Pada Sistem Informasi Pemasaran di Toko YEN-YEN)," *J. Sist. Inf.*, vol. 4, pp. 515–529, 2012.
- [12] M. R. Ravi Kalakota, *e-Business 2.0: Roadmap for Success*, 2nd Editio. Addison-Wesley Professional, 2000.
- [13] G. Egri and C. Bayrak, "The role of search engine optimization on keeping the user on the site," *Procedia Comput. Sci.*, vol. 36, no. C, pp. 335–342, 2014.
- [14] S. C. Vermaat, *Discovering Computers Fundamentals*, Third Edit. 2008.
- [15] Ibrahim N., "Pengembangan Aplikasi Semantic Web

- Untuk Membangun Web yang Lebih Cerdas,” *J. Inform.*, vol. 3, pp. 27–39, 2007.
- [16] M. A. N. dan H. K. F. S. Aghaei, “Evolution of The World Wide Web: From Web 1.0 to Web 4.0,” *Int. J. Web Semant. Technol.*, vol. 3, 2012.
- [17] Tim Berners-Lee, “The World Wide Web: A very short personal history,” 1998. .
- [18] T. Berners-Lee, *The semantic web*. scientific american.com, 2001.
- [19] Harris Saputra, *Pengelolaan Database MySQL dengan PHP MyAdmin*. Graha Ilmu, 2008.
- [20] Furrar Utdirartatmo, *Mengelola Database Server MySQL Di Linux Dan Windows*. yogy, 2012.
- [21] E. Miller, “How Not To Sort By Average Rating,” *Evanmiller.org*, 2009. .
- [22] “reddit’s new comment sorting system,” 2009. .

(Halaman ini sengaja dikosongkan)

BIODATA PENULIS



Penulis bernama Ahmad Yusuf Fauzan Adimah yang lahir di kota Surabaya, Jawa Timur pada tanggal 11 Juli 1995. Penulis merupakan anak terakhir dari tiga bersaudara. Penulis telah menempuh pendidikan formal di SDN Klampis Ngasem 1 Surabaya, SMP Negeri 19 Surabaya, dan SMA Negeri 20 Surabaya. Pada tahun 2013 penulis di terima di Jurusan Sistem Informasi FTIF –

ITS Surabaya dan terdaftar dengan NRP 5213100133 dengan jalur SBMPTN. Selain kesibukan akademik, penulis juga mengikuti beberapa kegiatan kemahasiswaan seperti menjadi anggota futsal JSI, koordinator perlengkapan Information System Expo (ISE), panitia wisuda, sie dokumentasi ISICO dan SESINDO. Penulis juga pernah menjalani kerja praktik di PDAM Kota Surabaya selama 2 bulan.

Pada akhir semester di Jurusan Sistem Informasi, penulis mengambil bidang minat Sistem Enterprise untuk tugas akhir. Untuk kepentingan penelitian, penulis juga dapat dihubungi melalui email *ahmadyusuffauzan@gmail.com*.

(Halaman ini sengaja dikosongkan)

LAMPIRAN A

```
Route::get('/home', 'UcanController@index');  
Route::get('/awal', 'UcanController@show');  
Route::get('/ups1', 'UcanController@ups1');  
Route::get('/downs1', 'UcanController@downs1');
```

Gambar A1 Route Sistem Pada Aplikasi

```
<?php  
// // include('koneksi.php');  
// $sql = "SELECT distinct te  
// // $result = $dbh->quer  
?>  
<table id="tabelku">  
  <thead>  
    <th>title</th>  
    <th>rating</th>  
    <th>ratingups</th>  
    <th>ratingdowns</th>  
    <th>id</th>  
    <th>action</th>  
  </thead>
```

Gambar A.2 Kode Tabel Pada Aplikasi

```

<?php foreach($result as $r1){ ?>
<tbody>
  <td><?php echo $r1->title; ?></td>
  <td id="rating-= $r1-&gt;id22; ?&gt;"&gt;&lt;?php echo $r1-&gt;rating; ?&gt;&lt;/td&gt;
  &lt;td name="ups" id="ups-<?= $r1-&gt;id22; ?&gt;"&gt;&lt;?php echo $r1-&gt;ups; ?&gt;&lt;/td&gt;
  &lt;td name="downs" id="downs-<?= $r1-&gt;id22; ?&gt;"&gt;&lt;?php echo $r1-&gt;downs; ?&gt;&lt;/td&gt;
  &lt;td name="id1" id="id1"&gt;&lt;?php echo $r1-&gt;id22; ?&gt;&lt;/td&gt;

&lt;/td&gt;
</pre

```

Gambar A.3 Kode Untuk Menampilkan Tabel

```

<!-- <input type="text" name="id1" id="id1" value="<?php echo $r1->id22; ?>"-->
<!-- <input type="text" name="ups" id="ups" value="<?= $r1->ups; ?>"-->
<!-- <input type="text" name="downs" id="downs" value="<?= $r1->downs; ?>"-->
  <button onClick="updateForm(= $r1-&gt;id22; ?&gt;)"&gt; up &lt;/button&gt;
  &lt;button onClick="updateForma(<?= $r1-&gt;id22; ?&gt;)"&gt; down &lt;/button&gt;
&lt;/td&gt;
&lt;/tbody&gt;
&lt;?php
}
?&gt;

&lt;/table&gt;
&lt;?php
?&gt;
</pre

```

Gambar A.4 Kode Untuk Tombol Action Up dan Down

```
<!DOCTYPE html>
<html>
<head>

  <link rel="stylesheet" type="text/css" href="css/jquery.dataTables.css">
  <title>Fitur Voting</title>

<style>
table {
  font-family: arial, sans-serif;
  border-collapse: collapse;
  width: 100%;
}

td, th {
  border: 1px solid #dddddd;
  text-align: left;
  padding: 8px;
}

tr:nth-child(even) {
  background-color: #dddddd;
}
</style>
```

Gambar A.5 Kode Untuk CSS home.blade.php

```

</head>
<body>
<div id="isitabel">
  <br>
  <table id="tabelku">
    <thead>
      <th>title</th>
      <th>rating</th>
      <th>ratingups</th>
      <th>ratingdowns</th>
      <th>id</th>
      <th>action</th>
    </thead>

    <?php foreach($result as $r1){ ?>
    <tbody>
      <td><?php echo $r1->title; ?></td>
      <td id="rating-<?=$r1->id22; ?>"><?php echo $r1->rating; ?></td>
      <td name="ups" id="ups-<?=$r1->id22; ?>"><?php echo $r1->ups; ?></td>
      <td name="downs" id="downs-<?=$r1->id22; ?>"><?php echo $r1->downs; ?></td>
      <td name="id1" id="id1"><?php echo $r1->id22; ?></td>

    <td>

```

Gambar A.6 Tabel Pada Aplikasi

```

<!-- <input type="text" name="id1" id="id1" value="<?php echo $r1->id22; ?>"-->
<!-- <input type="text" name="ups" id="ups" value="<?=$r1->ups; ?>"-->
<!-- <input type="text" name="downs" id="downs" value="<?=$r1->downs; ?>"-->
  <button onClick="updateForma(<?=$r1->id22; ?>)" up </button>
  <button onClick="updateForma(<?=$r1->id22; ?>)" down </button>
  </td>
</tbody>
<?php
}
?>

</table>
</div>

<div id="content"></div>
<!-- <script src="https://cdnjs.cloudflare.com/ajax/libs/jquery/3.2.1/jquery.min.js"><
</body>

```

Gambar A.7 Kode Untuk Menampilkan Tabel

```

<script src="{ url( 'js/jquery-2.2.4.min.js' ) }"></script>
<script type="text/javascript" charset="utf8" src="js/jquery.dataTables.js"></script>

<script>

function confidence(ups, downs){
    var n = ups + downs;
    // console.log(ups);
    // console.log(downs);
    if(n == 0){
        console.log('masuk');
        return 0;
    }else{
        var z = 1.96;
        var phat = 1.0*ups / n;
        var hasil;
        // var akan = (phat*(1-phat)+(z*z/(4*n))/n);
        hasil = (((phat + z*z/(2*n)) - (z * Math.sqrt((phat*(1-phat)+(z*z/(4*n))/n)))/(1+z*z/n));

        console.log(hasil);
        return hasil;
    }
}
</script>

```

Gambar A.8 Kode Penulisan Rumus Wilson Score Interval

```
function updateForm(myId) {  
  
    var id22 = myId;  
    var ups1 = document.getElementById('ups-'+myId);  
    var ups = parseFloat(ups1.innerHTML)+1.0;  
    console.log(ups);  
    var downs1 = document.getElementById('downs-'+myId);  
    var downs = parseFloat(downs1.innerHTML);  
    console.log(downs);  
  
    var hasil_conf = confidence(ups,downs);  
    var url_do = "ups1";  
    console.log( );  
    $.get(url_do,{  
  
        hasil_conf: hasil_conf,  
        id22 : id22,  
        ups : ups  
    }, function (data ) {  
  
        $("#isitabel").empty();  
        $("#isitabel").html(data);  
    });  
}  
</script>
```

Gambar A.9 Action Untuk Tombol Up

```
<script type="text/javascript">
    function updateForma(myId) {
        var id22 = myId;
        var ups1 = document.getElementById('ups-'+myId);
        var ups = parseFloat(ups1.innerHTML);
        var downs1 = document.getElementById('downs-'+myId);
        var downs = parseFloat(downs1.innerHTML)+1;

        var hasil_conf = confidence(ups,downs);
        console.log(parseFloat(hasil_conf));
        var url_do = "downs1";
        $.get(url_do,{

            hasil_conf: hasil_conf,
            id22 : id22,
            downs : downs
        }, function (data) {

            $("#isitabel").empty();
            $("#isitabel").html(data);
        });
    }
</script>
</html>
```

Gambar A.10 Action Untuk Tombol Down

```

</php>

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use Illuminate\Support\Facades\DB;

use App\application;

class UcanController extends Controller
{
    public function index(){

        $ucan = DB::select("SELECT distinct tes.id22, tes.id, tes.rating, applications.title, tes.ups, tes.downs
FROM tes INNER JOIN applications ON tes.id = applications.id where applications.genreId = 'music_and_audio'
AND (tes.id22 = 212 OR tes.id22 = 208 OR tes.id22 = 249 OR tes.id22 = 207 OR tes.id22 = 205 OR tes.id22 =
222 OR tes.id22 = 279 OR tes.id22 = 260 OR tes.id22 = 223 OR tes.id22 = 246 ) order by tes.rating desc,
applications.maxInstalls desc, applications.score desc, applications.reviews desc");

        return view('home',['result' => $ucan]);
    }
}

```

Gambar A.11 Controller Untuk Menampilkan Halaman Home

```

public function upsi(){
    $hasil_conf = $_GET['hasil_conf'];
    $id22 = $_GET['id22'];
    $ups = $_GET['ups'];
    $ups = $ups;
    $master = DB::table('tes')->where('id22', $id22)->update(['ups' => $ups,'rating'=> $hasil_conf]);
    $ucan = DB::select("SELECT distinct tes.id22, tes.id, tes.rating, applications.title, tes.ups, tes.downs FROM
tes INNER JOIN applications ON tes.id = applications.id where applications.genreId = 'music_and_audio' AND
(tes.id22 = 212 OR tes.id22 = 208 OR tes.id22 = 249 OR tes.id22 = 207 OR tes.id22 = 205 OR tes.id22 = 222 OR
tes.id22 = 279 OR tes.id22 = 260 OR tes.id22 = 223 OR tes.id22 = 246 ) order by tes.rating desc,
applications.maxInstalls desc, applications.score desc, applications.reviews desc");

    return view('awal',['result' => $ucan]);
}

```

Gambar A.12 Untuk Mengupdate Database Ketika Tombol Up Ditekan

```

public function downs1(){
    $hasil_conf = $_GET['hasil_conf'];
    $id22 = $_GET['id22'];
    $downs = $_GET['downs'];
    $downs = $downs;
    $master = DB::table('tes')->where('id22', $id22)->update(['downs' => $downs,'rating'=> $hasil_conf]);
    $ucan = DB::select("SELECT distinct tes.id22, tes.id, tes.rating, applications.title, tes.ups, tes.downs FROM
tes INNER JOIN applications ON tes.id = applications.id where applications.genreId = 'music_and_audio' AND
(tes.id22 = 212 OR tes.id22 = 208 OR tes.id22 = 249 OR tes.id22 = 207 OR tes.id22 = 205 OR tes.id22 = 222 OR
tes.id22 = 279 OR tes.id22 = 260 OR tes.id22 = 223 OR tes.id22 = 246 ) order by tes.rating desc,
applications.maxInstalls desc, applications.score desc, applications.reviews desc");

    return view('awal',['result' => $ucan]);
}
}

```

Gambar A.13 Untuk Mengupdate Database Ketika Tombol Down Ditekan

(Halaman ini sengaja dikosongkan)

LAMPIRAN B

Tabel B.1 Bukti Voting Pada Fase Pertama dan Kedua

Nama	NRP	Fase 1	Fase 2
Fitri Larasati	5213100175	- Smule Sing (1 up) - Shazam (1 up) - Soundhound (1 up)	- Deezer (1 up) - Synthesia (1 up) - Soundhound (1 up) - My Mixtapez (1 down)
Tetha Valianta	5213100055	- Tunein Radio (1 up) - Sound CLOUD (1 up) - Smule (1 up)	- smule (1 up) - sound cloud (1 up) - sound hound (1 up) - shazam (1 down)
Pramita Lucianna	5213100186	- soundcloud (1 up) - smule (1 up) - music player (1 up)	- syntesia (1 up) - deezer (1 up) - shazam (1 up) - ares musician (1 down)
Harun Rizal	5213100125	- tune inn (1 up) - Soundcloud (1 up) - Shazam (1 up) - Soundhound (1 up)	- soundhound (1 up) - soundcloud (1 up) - deezer (2 up) - ares (1 down)

		- Smule (1 down)	
Bintang Setyawan	5213100011	- soundcloud (1 up) - soundhound (1 up) - Tuneinn (1 up)	- soundhound (1 up) - soundcloud (1 up) - tune in (1 up) - my mixtapez (1 up) - ares (1 down)
Yeremia Aha	5213100135	- soundcloud (1 up) - my mixtapez (1 up) - music player (1 up)	- smule (1 up) - mixtape(1 down) - shazam (1 down) - ares (1 down)
Frizka Izza	521310003	- smule (1 up) - soundcloud (2 up) - shazam (1 up) - deezer (1 down)	- deezer (1 up) - soundcloud (1 up) - soundhound (1 up) - smule (1 down) - mixtape (1 down) - shazam (1 down)
Nanda Puji	5213100023	- soundhound (1 up) - music player (1 up) - soundcloud (1 up)	- soundcloud (1 up) - soundhound (1 up) - deezer (1 up) - synthesia (1 down)

			- ares (1 down)
Alvin Rahman	5213100031	- soundcloud (1 up) - soundhound (1 up) - deezer (1 up)	- soundcloud (1 up) - soundhound (1 up) - deezer (1 up)
Andika Aji	5213100013	- soundhound (1 up) - deezer (1 up) - synthesia (1 up)	- deezer (1 up) - my mixtape (1 up) - music player (1 up) - soundcloud (2 down) - smule (1 down) - tune in (1 up) - syntehsia (1 up)

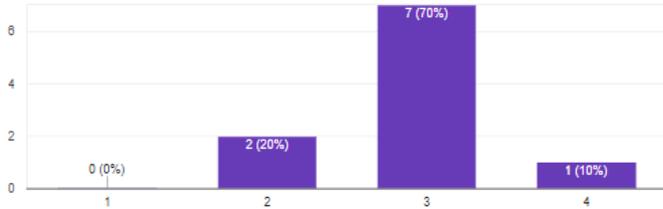
(Halaman ini sengaja dikosongkan)

LAMPIRAN C

Peer Review 01

Faktor waktu mempengaruhi perubahan pilihan yang anda lakukan pada fase pertama ke fase kedua

10 responses



Gambar C.1 Hasil Review Perspektif Pengguna

*Jika tidak setuju berikan alasannya

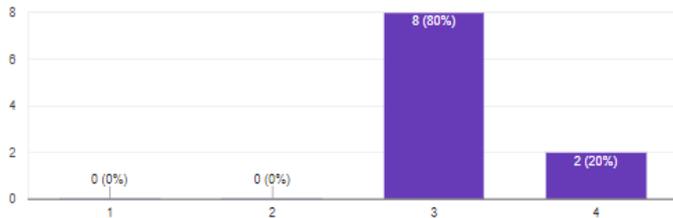
2 responses

karena faktor selera dan informasi fitur adalah faktor yg menentukan
karena yang mempengaruhi perubahan pilihan saya adalah ketika saya mencoba aplikasi

Gambar C.2 Alasan Pengguna Bila Tidak Setuju

Faktor knowledge mempengaruhi perubahan pilihan yang anda lakukan pada fase pertama ke fase kedua

10 responses



Gambar C.3 Hasil Review Pengguna Faktor Knowledge

Peer Review 02

Dari kedua urutan rangking tersebut, manakah urutan rangking yang paling sesuai menurut anda

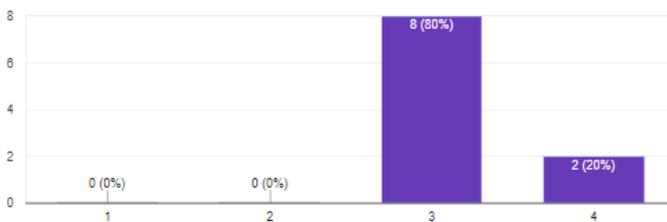
10 responses



Gambar C.4 Hasil Review Perspektif Pengguna Dari Kedua Metode

Seberapa tepatkah hasil urutan rangking menurut pendapat anda

10 responses

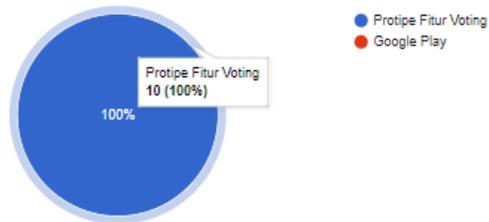


Gambar C.5 Hasil Review Perspektif Pengguna Ketepatan Hasil Urutan Rangking

Peer Review 03

Dari kedua urutan ranking tersebut, manakah urutan ranking yang paling sesuai menurut anda

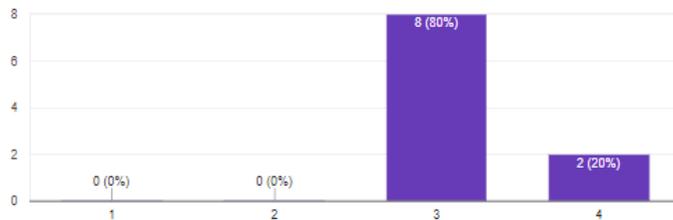
10 responses



Gambar C.6 Hasil Review Perspektif Pengguna Prototipe dengan GooglePlay

Seberapa tepatkah hasil urutan ranking menurut pendapat anda

10 responses



Gambar C.7 Hasil Review Perspektif Pengguna Ketepatan Hasil Urutan Ranking